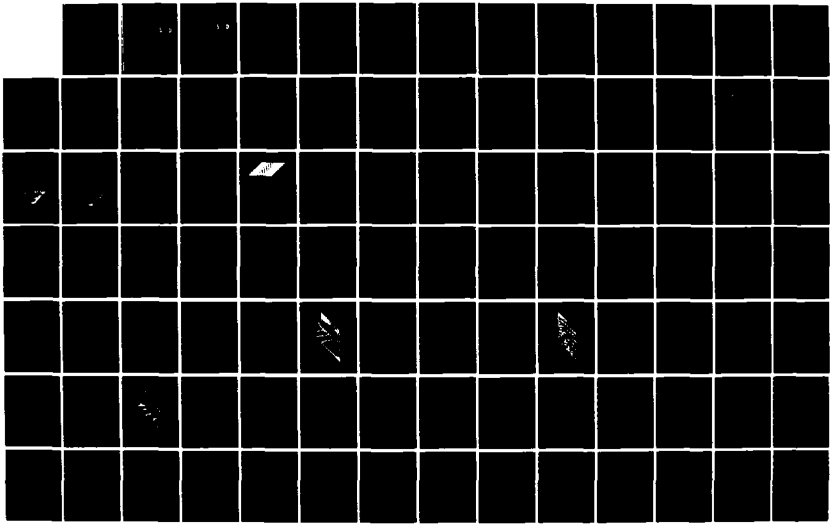
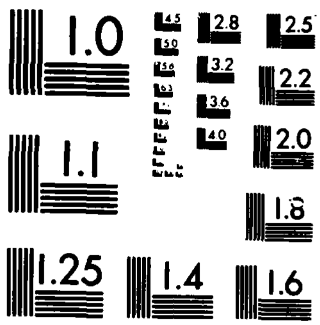
AD-A163 953 EFFECTS OF RANDOM PHASE SHIFTS AND CARRIER DRIFT ON THE RESOLUTION PROPER..(U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI.. T L GRIFFIN 1/2

UNCLASSIFIED DEC 85 AFIT/GE/ENG/85D-19 F/G 9/4 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

1

EFFECTS OF RANDOM PHASE SHIFTS AND CARRIER DRIFT ON THE RESOLUTION PROPERTIES OF LINEAR STEP FREQUENCY PULSE TRAINS

THESIS

Thomas L. Griffin, Jr.
Captain, USAF

AFIT/GE/ENG/85D-19

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GE/ENG/85D-19

EFFECTS OF RANDOM PHASE SHIFTS AND CARRIER
DRIFT ON THE RESOLUTION PROPERTIES OF
LINEAR STEP FREQUENCY PULSE TRAINS

THESIS

Thomas L. Griffin, Jr.
Captain, USAF

AFIT/GE/ENG/85D-19

AFIT/GE/ENG/85D-19

EFFECTS OF RANDOM PHASE SHIFTS AND CARRIER DRIFT ON THE RESOLUTION PROPERTIES OF LINEAR STEP FREQUENCY PULSE TRAINS

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Electrical Engineering

Accesion For		
NTIS CRA&I	☑	
DTIC TAB	☐	
Unannounced	☐	
Justification		
By		
Distribution /		
Availability Codes		
Dist	Avail and / or Special	
A-1		

Thomas L. Griffin, Jr., B.S.

Captain, USAF

December 1985

Approved for public release; distribution unlimited

Preface

The purpose of this study is to determine the effects of random initial phases and random carrier drift on the range and doppler resolution of linear step frequency pulse trains. The method is based on computing the pulse train's ambiguity function, which describes the response of a matched filter to the waveform as a function of delay and doppler shift.

Chapter I provides motivation for studying the effects of phase and carrier drift, as well as refinement of the scope and limitations of this study. Chapter II presents a brief overview of ambiguity function theory, and develops specific expressions for the linear step frequency waveform, including effects of random phases and carrier drift. The major effort of this study has been the development of the computer software to calculate, tabulate, and plot the ambiguity function expressions developed. A discussion of the programs is presented in Chapter III. The software is validated in Chapter IV by presenting results consistent with those known for the constant carrier pulse train and predicted for the linear step frequency pulse train. The effects of random phases and carrier drift on the resolution properties are then presented in Chapter V. Conclusions and recommendations are given in Chapter VI.

I express my appreciation and gratitude to Dr. Vittal Pyati, my thesis advisor, for the guidance given and the patience shown during this effort. I thank the members of my thesis committee for the help

each has provided. I commend the staff of the Signal Processing Laboratory, especially Capt David King and Mr. Dan Zambon, for the excellent computer support, and I thank my fellow students for their support and friendship.

To Christy, my wife and life companion, I shall eternally be grateful for the special love and support given me the last 18 months. Finally, I thank my Lord Jesus Christ, for without His help, none of this would have been possible.

Thomas L. Griffin, Jr.

Table of Contents

List of Figures

Figure | Page

List of Tables

Abstract

A FORTRAN program using fast Fourier transforms has been written to calculate and plot the ambiguity function of a uniform linear step frequency pulse train with random initial phases and carrier frequency drifts. The effects of random phases and frequency drifts are studied in two stages. First, the initial phases are allowed to assume any value with equal probability in ranges of 0 to 20 degrees and 0 to 180 degrees while the frequency drifts are zero. Secondly, the phases are fixed at zero and carrier frequency drifts up to 0.1% and 1.0% are set. In each case, the ambiguity surface is shifted along the doppler axis with the effect being more pronounced for carrier frequency drifts. The effects on the half power width of the central lobe for 10 test cases for each phase and frequency range appear negligible for a two pulse train.

[illegible]

EFFECTS OF RANDOM PHASE SHIFTS AND CARRIER DRIFT ON THE RESOLUTION PROPERTIES OF LINEAR STEP FREQUENCY PULSE TRAINS

I. Introduction

Statement of the Problem

An important and frequently implemented class of transmitted radar waveforms is the coherent pulse train. This class of waveforms has the desirable property of increasing the time-bandwidth product, thereby improving the range and range rate or doppler resolution, but simultaneously introduces undesirable ambiguities in both range and doppler measurements. A host of variations on the basic uniformly spaced coherent pulse train have been described in order to maximize the resolution properties and reduce the effect of the range and range rate ambiguities (Ref 3, Ref 7 and Ref 11).

The ambiguity function introduced by Woodward is frequently used to visualize the resolution properties of a specific waveform (Ref 15: Chap 7). Studies by Rihaczeck (Ref 11) and Cook and Bernfeld (Ref 3) are frequently referenced for describing the ambiguity function and ambiguity surface plots of specific waveforms. In plotting and otherwise depicting ambiguity surfaces for pulse trains, initial phase and carrier frequency coherence is almost always assumed. However, in practice, the radar systems engineer desires to know both qualitatively and quantitatively the consequence on the resolution properties of waveforms if

coherence is not maintained.

This study investigates the effects of noncoherence on the delay and doppler resolution properties of a specific type of pulse train: linear, interpulse frequency shift coding of uniformly spaced and identically shaped rectangular pulses. The effects of noncoherence are investigated for pulse to pulse, random initial phase differences in ranges of both 20 and 180 degrees, and for pulse to pulse random carrier frequency drift in ranges of 0.1 and 1.0 percent.

Background

"The radar systems engineer can count, along with the finite human span and taxes, distortion and its undesirable effects as being among the certainties" (Ref 3:366). Given then that distortion is inevitable and must be dealt with, the radar systems engineer must decide what level of distortion is tolerable. The ability to visualize distortion's effects at various levels would be desirable. A plot of the ambiguity function provides that ability for the output of a matched filter receiver.

During World War II, radar engineers realized that improved radar performance was presently limited by transmitter power output. Since peak power output of transmitter tubes was limited, an obvious solution was to obtain higher average power by using wider pulse widths. This approach was, however, in conflict with the simultaneous requirement for greater resolution of targets. A solution to resolve this apparent dilemma was proposed in which a wide pulse is transmitted, during which

the carrier frequency is linearly swept over some range of frequencies. The radar receiver incorporates a pulse compression filter having a linear time versus frequency relation such that the beginning of the received pulse is delayed relative to its end by an appropriate amount. The output is a waveform compressed in time and increased in amplitude (Ref 3:10). Such a radar system has the desired capabilities of high average power due to transmission of a wide pulse and increased resolution due to the receiver's compression of the received pulse.

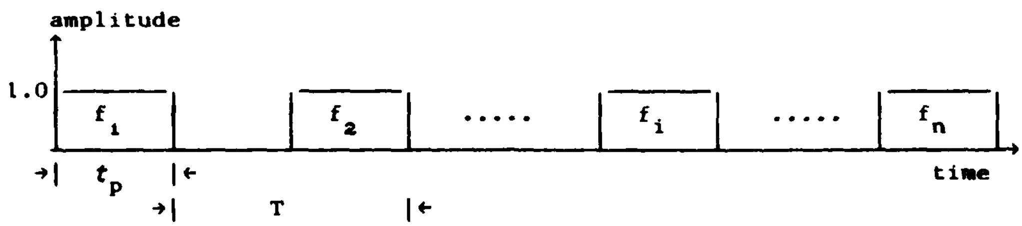
In addition to requiring high power output to increase radar performance, radar engineers strived to maximize the received signal to noise ratio. The receiver which achieves this end for additive white noise was derived by North and is call the "matched filter" for the transmit waveform (Ref 7:278). The derivation of this filter's transfer function and impulse response is well known, and will not be presented here. However, a word picture description of the impulse response is appropriate as it will lead to the ambiguity function: the impulse response of a matched filter is a time-reversed image of the waveform to which the filter is matched, multiplied by a constant. The output of the matched filter is the convolution of the input waveform and the filter's impulse response. The output may alternatively be described as a time-shifted replica of the autocorrelation of the input waveform (Ref 7:280).

Soon after its introduction by Woodward, ambiguity function analysis became a popular method of investigating a waveform's resolution potential. One definiton of the ambiguity function is

$$\chi(\tau,\nu) = \int_{-\infty}^{+\infty} \mu(t)\mu^*(t-\tau)e^{j2\pi\nu\tau}dt \qquad (1)$$

where τ is time delay, ν is doppler shift, $\mu(t)$ is the complex envelope of the transmitted signal, and * indicates the complex conjugate operation. It represents the response of the matched filter to the signal for which it is matched as well as to doppler-frequency-shifted (mismatched) signals (Ref 13:411), or alternatively may be viewed as the two-dimensional correlation function in delay and doppler (Ref 11:70).

Woodward showed that range resolution and accuracy are functions not of transmitted pulse width as had been believed, but of the transmitted signal bandwidth. Many waveforms have been proposed which take advantage of this fact, and are generally classified as pulse compression waveforms. One such waveform, and that one to be studied herein, is the discrete, linear step frequency, uniformly spaced pulse train depicted in Figure 1.



t_p = pulsewidth   T = repetition period

f_i = transmitted frequency = $f_o + n\Delta f$, where f_o is the nominal carrier frequency, n is an integer, and Δf is the frequency step size.

Figure 1. Step-Frequency Pulse Train

As mentioned in the preceding section, pulse trains generally exchange increased range and doppler resolution capabilities for range

and doppler ambiguities. However, the linear step frequency waveform depicted in Figure 1 could completely eliminate major range ambiguities due to multiple-time-around echoes (Ref 11:318), and thereby warrants additional study.

Assumptions

This study analyzes the selected pulse train under the assumptions of a simplified resolution theory for range and range rate. That theory asserts that for waveforms with time-bandwidth products less than $0.1c/\dot{R}$ where c is the speed of light and $\dot{R}$ the maximum target range rate of interest, the complex envelope of the transmitted waveform is not significantly distorted by target motion, and that the received envelope, therefore, may be considered simply a time delayed replica of the transmitted envelope. In general, the time-bandwidth product of pulse trains violates this simplification, necessitating a more complicated analysis. However, for restricted values of range rate, this simplification holds (Ref 11 :60-61,289). Additionally, this study assumes that the rectangular pulses have instantaneous rise-times, constant pulse widths, constant pulse repetition times, and assumes that the return pulse amplitudes are not a function of time.

Limitations

The investigation of the effects of phase differences and drift in carrier frequency is limited to discrete, pulse to pulse random phase

differences and random carrier frequency drifts, as opposed to functions of time. This limitation is imposed in order to simplify calculations in computing the ambiguity function. Due to time constraints, limitations are also placed on the range that phase differences and carrier drifts may assume. The effects of two ranges, 20 and 180 degrees, of phase differences are examined, while carrier drifts are limited to 0.1 and 1.0 percent.

Presentation

This study presents first a derivation of the general form of the ambiguity function, and then derives two specific forms for a discrete, linear step frequency, uniformly spaced, rectangular pulse train. The first form represents the zero initial phase and zero carrier frequency drift case. The second form incorporates both random initial phase terms and random carrier drifts in each pulse of the waveform.

Computer software is developed to calculate, tabulate, and plot the magnitude of the ambiguity function for the waveform. Comparisons of range and doppler resolution capabilities of both forms for the two ranges of phase and carrier drift described above are then made based on the magnitude of the ambiguity function.

Finally, conclusions on the effects of random phases and carrier drifts are drawn, and recommendations for future study made.

II. Theoretical Considerations

The Ambiguity Function

Return signals from radar targets are essentially a modified version of the transmitted signal. The modifications, excluding obvious amplitude differences, may be considered to be of two types: a time delay (τ) proportional to the target range, and a frequency shift (ν) proportional to the target radial velocity. Similarily, differences in returns from two targets may be only a time delay and a frequency shift due to differences in their range and radial velocity. The differences in the returned signals enable radar systems to resolve multiple targets.

A measure of signal difference which leads to the ambiguity function is the integral square error, defined as:

$$\mathcal{E}^2 = \int_{-\infty}^{+\infty} \left[f(t)-g(t)\right]^2 dt \tag{2}$$

where f(t) and g(t) represent two real signals. In terms of complex representation, the difference is

$$\mathcal{E}^2 = \frac{1}{2}\int_{-\infty}^{+\infty} \left|f_a(t)-g_a(t)\right|^2 dt \tag{3}$$

where $f_a(t)=f(t)+j\hat{f}(t)$, $g_a(t)=g(t)+j\hat{g}(t)$, and $\hat{f}(t)$ and $\hat{g}(t)$ are the Hilbert transforms of f(t) and g(t) (Ref 2:10). Expanding Eq. 3 yields

$$2\mathcal{E}^2 = \int_{-\infty}^{+\infty} \left| f_a(t) \right|^2 dt + \int_{-\infty}^{+\infty} \left| g_a(t) \right|^2 dt - 2\int_{-\infty}^{+\infty} \mathrm{Re}\left\{ f_a(t) g_a^*(t) \right\} dt \qquad (4)$$

where * indicates the complex conjugate. This derivation assumes the the transmitted signal s(t) is of the form $a(t)\cos[w_o t+\phi(t)]$, and that the carrier w_o is high so that the spectrum of s(t) contains negligible low frequency energy. Then s(t) may be represented as

$$s(t) = \mathrm{Re}[\varphi(t)] = \mathrm{Re}\left[\mu(t) e^{jw_o t}\right]$$

where $\varphi(t)$ and $\mu(t)$ are the complex representation and the complex envelope s(t). Then, letting f(t) be the transmitted signal s(t), and g(t) be a delayed, doppler shifted version $\mu(t-\tau)e^{-j(w_o+w_d)(t-\tau)}$, Eq. 3 becomes

$$2\mathcal{E}^2 = \int_{-\infty}^{+\infty} \left| \mu(t) \right|^2 dt + \int_{-\infty}^{+\infty} \left| \mu(t-\tau) \right|^2 dt$$

$$- \quad 2\ \mathrm{Re} \left\{ \int_{-\infty}^{+\infty} \mu(t) e^{jw_o t} \mu(t-\tau) e^{-j(w_o+w_d)(t-\tau)} dt \right\} \qquad (5)$$

After collecting terms in the last integral on the right, and since

$$\int_{-\infty}^{+\infty} \left| \mu(t) \right|^2 dt = \int_{-\infty}^{+\infty} \left| \mu(t-\tau) \right|^2 dt = 2\int_{-\infty}^{+\infty} f^2(t) dt$$

(Ref 2:11), Eq. 5 becomes

$$\mathcal{E}^2 = 2\int_{-\infty}^{+\infty} f^2(t) dt - \mathrm{Re}\left\{ e^{j(w_o+w_d)\tau} \int_{-\infty}^{+\infty} \mu(t) \mu^*(t-\tau) e^{jw_d t} dt \right\} \qquad (6)$$

The last integral on the right then is of the form of the ambiguity function defined in Eq. 1 and repeated here as

$$\chi(\tau,\nu)=\int_{-\infty}^{+\infty}\mu(t)\mu^*(t-\tau)e^{j2\pi\nu t}dt$$

The ambiguity function of a signal can then be described as the complex envelope of the signal and the output of a matched filter receiver (Ref 14: 3-8), or equivalently as the correlation of the transmitted signal with the delayed, doppler shifted return signal. In $\chi(\tau,\nu)$, τ represents the difference between any part of the signal and the delay corresponding to the target range, and ν is the difference between the specific doppler shift of the signal and the doppler shift to which the receiver filter is matched (Ref 12:154). τ and ν are related to range (R) and velocity (V) as follows:

$$\tau = \frac{2R}{c} \quad \text{and} \quad \nu = \frac{2f_oV}{c} \tag{7}$$

where c is the velocity of propagation and f_o is the carrier frequency.

Properties of the Ambiguity Function

In this section, several important properties of the ambiguity function are presented which will later enable a preliminary evaluation of this study's calculation and plot of the ambiguity function for the linear, step-frequency waveform.

Assuming the energy in the complex envelope is normalized, then

$$\chi(0,0) = \int_{-\infty}^{+\infty} |\mu(t)|^2 dt = 1 \tag{8}$$

and

$$\chi(0,0) \geq |\chi(\tau,\nu)| \tag{9}$$

These properties state that the normalized ambiguity function has its maximum value at $\tau = \nu = 0$. By changing the signs of τ and ν in the defining equation, then it can be shown that

$$\chi(-\tau,-\nu) = e^{j2\pi\nu\tau}\,\chi(\tau,\nu)$$

However, since it is the magnitudes of these functions that are the quantities of interest, it is seen that

$$|\chi(-\tau,-\nu)| = |\chi(\tau,\nu)| \tag{10}$$

This relation reveals that the magnitude of the ambiguity function is symmetrical about the origin. The ambiguity function can also be shown to be its own Fourier transfrom (Ref 11:120). This leads to three important relations, the first of which is

$$\iint_{-\infty}^{+\infty} \left|\chi(\tau,\nu)\right|^2 d\tau d\nu = \left|\chi(0,0)\right|^2 = 1 \tag{11}$$

This relation means that the area under the normalized ambiguity surface, $|\chi(\tau,\nu)|^2$, is constant and independent of the specific waveform (Ref 11:122). The second and third relations are

$$\int_{-\infty}^{+\infty} \left|\chi(\tau,\nu)\right|^2 d\tau = \int_{-\infty}^{+\infty} \left|\chi(\tau,0)\right|^2 e^{-j2\pi\nu\tau} d\tau \qquad (12)$$

and

$$\int_{-\infty}^{+\infty} \left|\chi(\tau,\nu)\right|^2 d\nu = \int_{-\infty}^{+\infty} \left|\chi(0,\nu)\right|^2 e^{j2\pi\nu\tau} d\nu \qquad (13)$$

These relations state that the integrated volume distribution in doppler is determined by the value of the ambiguity function on the delay axis, and that the volume in delay is determined by the value on the doppler axis. A useful interpretation of these relations is that alteration of a waveform which reduces the volume of its ambiguity surface along one axis will increase the volume along the corresponding axis (Ref 11:122). Therefore, while reduction in volume along one axis will lead to an increase in resolution of the axis parameter, the volume increase along the opposite axis will introduce additional, perhaps undesired, response called self-clutter.

Classification of Waveforms by Ambiguity Function

As an alternative to classifying radar waveforms by the modulation function used, Rihaczek classified them according to their resolution potential as reflected by plotting their ambiguity function (Ref 10). Such waveforms are assigned to one of four classifications as presented in Table 1. The table lists the four classifications, describes the

basic waveform, and identifies the distinguishing feature of their ambiguity surface. A synopsis of the waveform classifications is presented so that in later sections the developement and plot of the ambiguity function for the linear, step frequency waveform can be analyized and predicted.

TABLE I.

Waveform Classification by Ambiguity Function

	Class A	Class B1	Class B2	Class C
Waveform type	Single pulses	Noise type	Linear FM	Coherent pulse trains
Ambiguity surface	Ridge	Thumbtack	Sheared-ridge	Bed-of-Nails

Class A waveforms are single, constant carrier pulses. As an example, Figure 2 depicts $|\chi(\tau,\nu)|$ for a rectangular pulse of duration T. Waveforms of this class have a time-bandwidth product (TB) of unity. The delay resolution is given by $\Delta\tau = 1/B$ and the doppler resolution by $\Delta\nu = 1/T$ (Ref 6:51). The distinguishing feature of this ambiguity surface is the ridge along the delay axis. Figure 3 depicts the details of the plot. In Figure 3(a), the triangular shape governed by $1-(|\tau|/T)$ at $\nu = 0$ is evident. Also evident is the (sin x)/x profile In Figure 3(b), the triangular shape is seen to degenerate as ν increases. The equation for the surface of Figure 2 is given by

$$|\chi(\tau,\nu)| = \left[1-\frac{|\tau|}{T}\right] \frac{\sin[\pi\nu T(1-|\tau|/T)]}{\pi\nu T(1-|\tau|/T)} \tag{14}$$

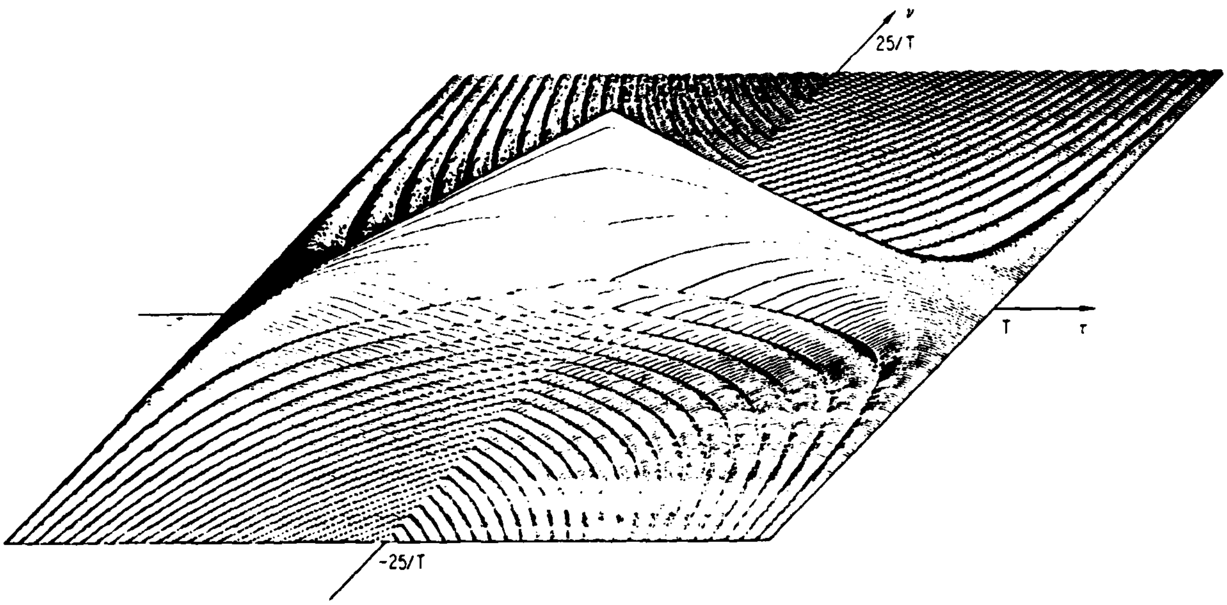


Figure 2. $|\chi(\tau,\nu)|$ of a Constant Carrier Pulse with a Rectangular Envelope (Ref 11:175).

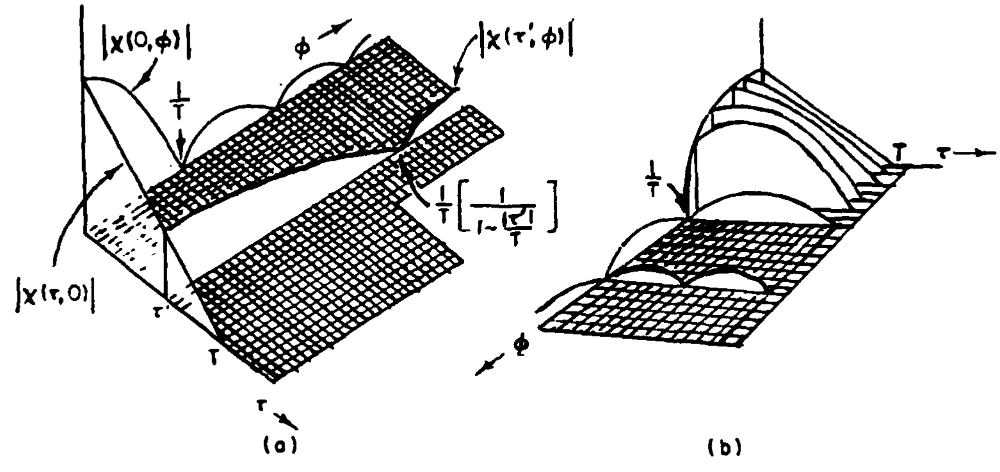


Figure 3. Time-Frequency Response for Rectangular Envelope Monotone Pulse. (a) Constant τ Profiles (b) Constant ν Profiles (Ref 3:93).

Class B1 is composed of pulse compression waveforms whose amplitude or frequency modulation function is irregular or noiselike. Types of waveforms in this class include non-linear frequency modulated (FM) pulses, long pulses with irregular amplitude modulation, binary shift codes (psuedo-random, Barker codes), poly-phase codes (Frank codes, Huffman codes), pulse trains with non-uniform or staggered PRF, and pulse trains with frequency shift coding (Ref 6:52). A Barker code waveform is a constant carrier rectangular pulse having N contiguous subpulses. Each subpulse may be in phase or 180 degrees out of phase with the preceding subpulse. The phase reversals of the pulse are ordered so that the autocorrelation will have a peak of unity and uniformly low sidelobes of magnitude 1/N (Ref 11:215). The distinguishing feature of the ambiguity surface of this class is its overall "inverted thumbtack" appearance. The 13 element Barker code waveform clearly reflects this appearance when $\nu = 0$ as in Figure 4. Figure 5 is a plot of $|\chi(\tau,\nu)|$ for the waveform.

Class B2 consists of linear FM waveforms. The waveforms' linear slope may be constant over the pulse duration, may be repeated two or more times, or may be composed of several slopes combined to form a complicated waveform. Figure 6 shows examples of the frequency slope of three linear FM waveforms. For the slope depicted in Figure 6(a) and a rectangular pulse, the ambiguity function is given by Eq. 14. The distinguishing feature of its plot is the orientation of the ridge along the line given by $\tau = -\nu T/B$, where T is the pulse duration and B is the frequency range swept. Figure 7 shows this "shearing" of the ridge.

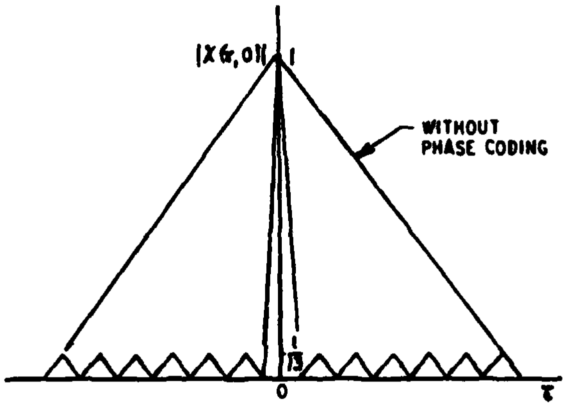


Figure 4. Autocorrelation Function of the 13 Element Barker Code (Ref 11:216).

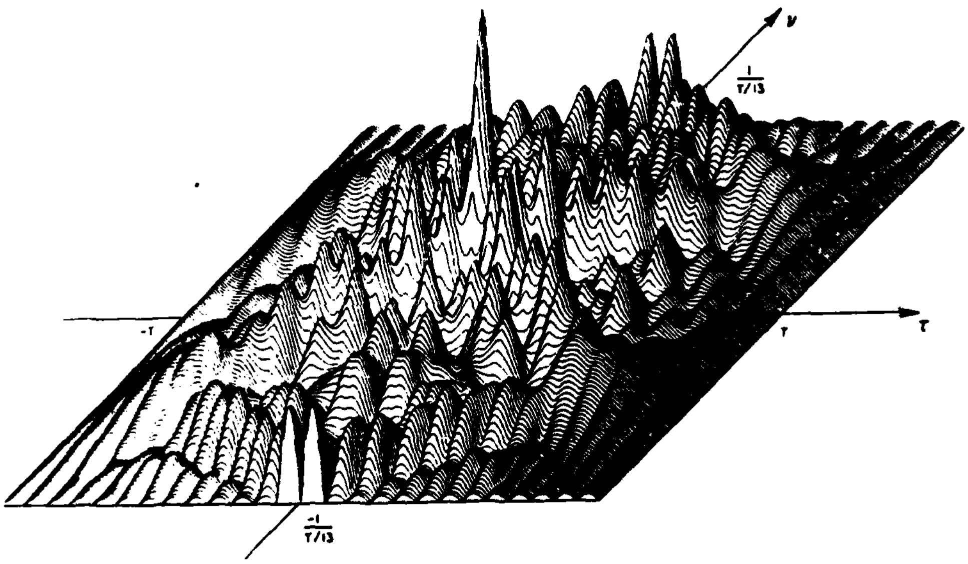


Figure 5. $|\chi(\tau,\nu)|$ for the 13 Element Barker Code (Ref 11:217).

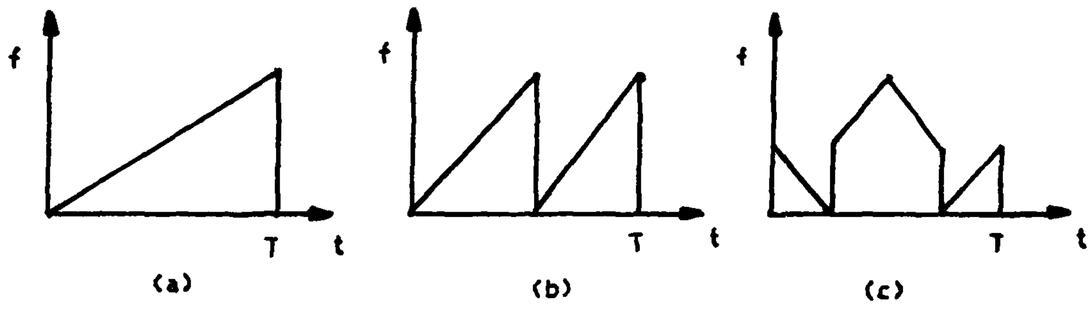


Figure 6. Class B2 Waveforms. (a) Linear FM
(b) Double Linear FM (c) Double V FM.

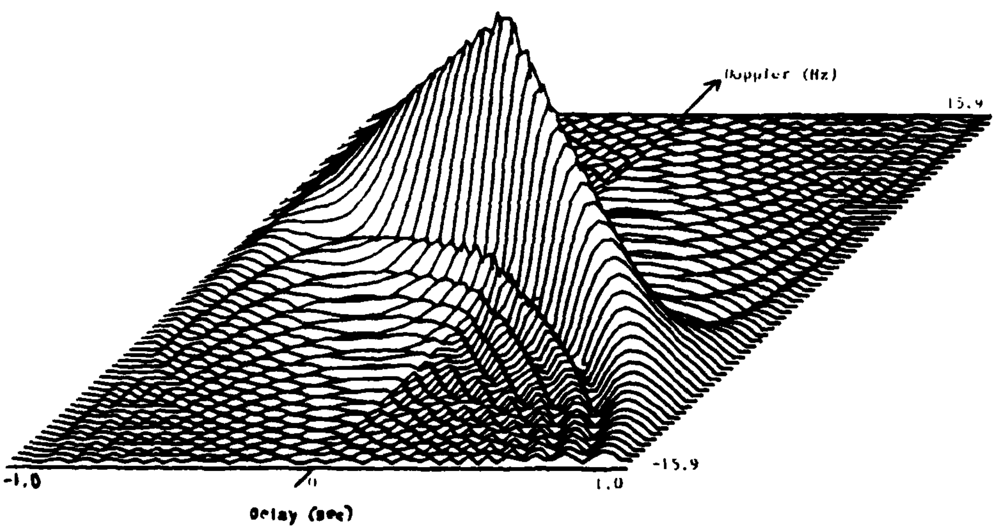


Figure 7. $|\chi(\tau,\nu)|$ for Linear FM (Ref 9:112).

The basic Class C waveform is the coherent pulse train whose ambiguity function is given by

$$\chi(\tau,\nu) = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} e^{j2\pi\nu nT} \chi_c\left[\tau-(m-n)T,\nu\right] \qquad (15)$$

where T is the pulse repetition period, N is the number of pulses in the waveform, and $\chi_c(\tau,\nu)$ is the ambiguity function of a single component pulse. This is the only class of waveform whose ambiguity surface may achieve a clear area in both delay and doppler around the central response peak (Ref 6:53). The plot of $\chi(\tau,\nu)$ is the superposition of the 2N ambiguity surfaces of the component pulse translated to positions $\tau = (m-n)T$ on the delay axis and weighted by a phase factor (Ref 11:185).

With expansion of the double sum, Eq. 15 can be written as

$$\chi(\tau,\nu) = \sum_{p=-(N-1)}^{N-1} e^{j\pi\nu(N-1+p)T} \chi_c(\tau-pT,\nu) \frac{\sin \pi\nu(N-|p|)T}{\sin \pi\nu T} \qquad (16)$$

In this form, the exponential in Eq. 15 can be seen to have a sampling like effect on the p=(n-m) surfaces in accordance with the function $\sin \pi\nu(N-|p|)T/\sin \pi\nu T$. It can also be shown that the peak amplitude of the p surfaces varies as $(N-|p|)/N$. Thus the peaks parallel to the doppler axis in each p surface will be lower and the variation slower for larger values of $|p|$. These characteristics can be seen in Figure 8 for a pulse train where N= 5 and the duty ratio $t_p/T = 0.3$, where t_p is the pulse width. On each side of the doppler axis there are N-1= 4 p surfaces, spaced in delay in integer multiples of the pulse repetition time T. Each p surface is seen to be sampled by $\sin \pi\nu(N-|p|)T/\sin \pi\nu T$, with

peaks at intervals of $\nu = kT$, where k is an integer. The total number of peaks in a p surface is the reciprocal of the duty ratio (T/t_p) as seen in the central p surface of Figure 9, where N=2, and $T/t_p = 4$. In Figure 8, it is seen that the doppler width of peaks in p surfaces increases with p, until, for the outermost p surfaces, the peaks merge and no breakup occurs. The doppler peak widths are given by $1/(N-|p|)t_p$ (Ref 11:293).

As a consequence of the pulse repetition, the resolution in range remains the same, but pronounced ambiguities occur at intervals of the repetition rate. However, doppler resolution does improve from 1/T to $1/Nt_p$, but ambiguities occur at interval of $1/t_p$. Overall, pulse repetition increases resolution potential at the expense of increased ambiguity.

<u>Representing the Linear Step Frequency Waveform</u>

In this section, mathematical expressions are developed to represent the linear step frequency waveform depicted in Figure 1, and a delayed, doppler shifted version of the waveform. These expressions are used to develop an expression for this waveform's ambiguity function.

The pulse train, s(t), of Figure 1 may be described as

$$s(t) = \sum_{n=0}^{N-1} \text{rect}\left[\frac{t-nT}{t_p}\right] \cos\left[2\pi(f_o + b_n f_o + nf_r)(t-nT) + \phi_n\right] \qquad (17)$$

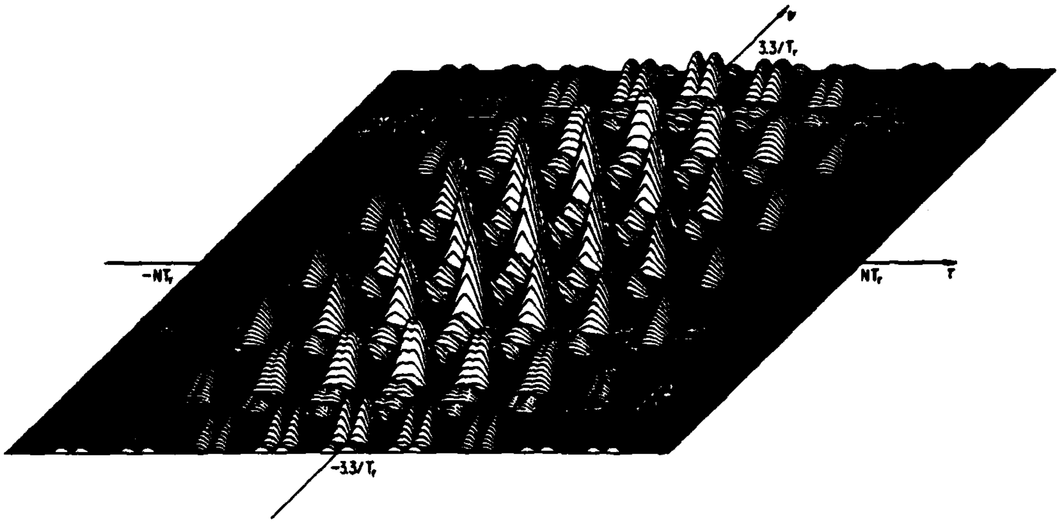


Figure 8. $|\chi(\tau,\nu)|$ for the Uniform Pulsetrain, N=5, Duty Ratio T/T_r= 0.3 (Ref 11:294).

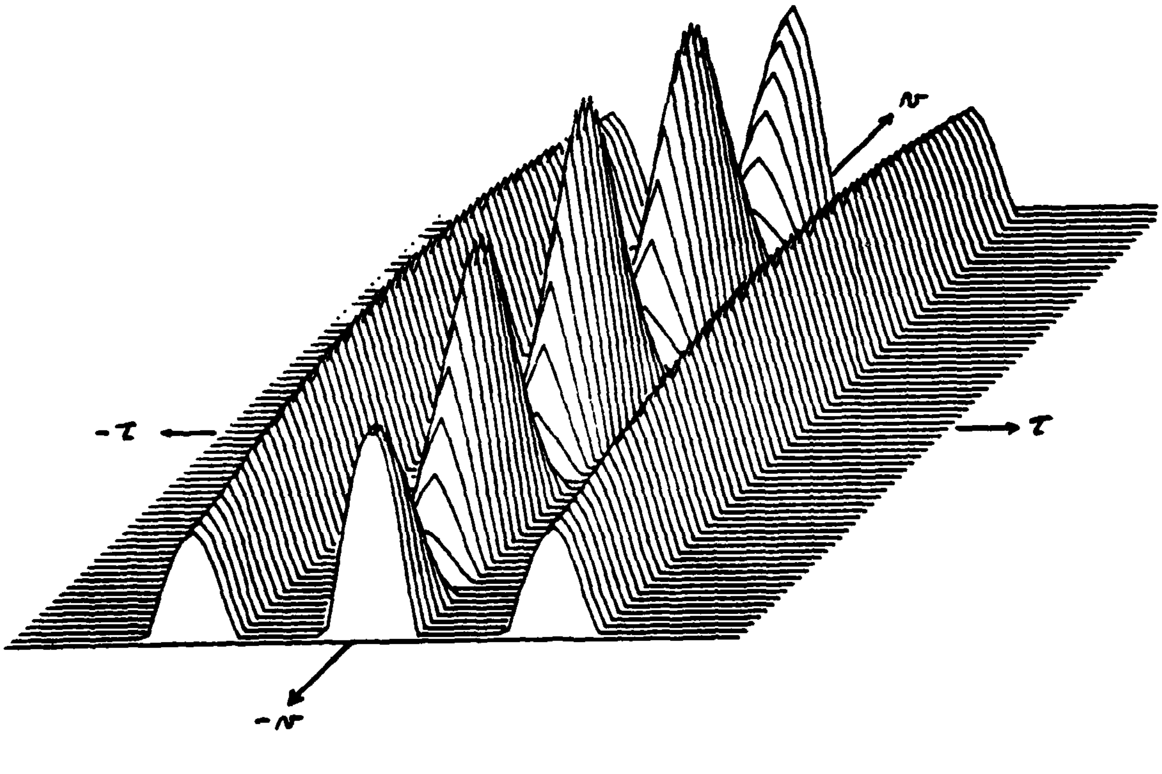


Figure 9. $|\chi(\tau,\nu)|$ for the Uniform Pulsetrain, N=2, T/t_p= 4.

where N is the number of pulses in the waveform, T is the pulse repetition period, f_o is the carrier frequency, $b_n f_o$ is carrier drift where b_n is a random percentage, f_r is the frequency step size, $\emptyset_n$ is initial phase, and $\mathrm{rect}\left[\frac{t}{\tau_p}\right] = 1$ for $0 \leq t \leq \tau_p$. In complex representation, the waveform may be expressed as

$$s(t) = \mathrm{Real}\left[\Psi(t)\right] \tag{18}$$

where

$$\Psi(t) = \sum_{n=0}^{N-1} \mathrm{rect}\left[\frac{t-nT}{\tau_p}\right] e^{j[2\pi(f_o+b_n f_o+nf_r)(t-nT)+\emptyset_n]}$$

Now define the complex envelope of s(t) as

$$\mu(t) = \Psi(t)e^{-j2\pi f_o t}$$

Then it follows that

$$\mu(t) = \sum_{n=0}^{N-1} \mathrm{rect}\left[\frac{t-nT}{\tau_p}\right] e^{j[2\pi(b_n f_o+nf_r)(t-nT)+\emptyset_n]} \tag{19}$$

and that a delayed form of $\mu(t)$ is

$$\mu(t-\tau) = \sum_{n=0}^{N-1} \mathrm{rect}\left[\frac{t-\tau-nT}{\tau_p}\right] e^{j[2\pi(b_n f_o+nf_r)(t-\tau-nT)+\emptyset_n]} \tag{20}$$

The Ambiguity Function for the Waveform

Recall that the ambiguity function as defined in Equation 1 is

$$\chi(\tau,\nu)=\int_{-\infty}^{+\infty}\mu(t)\mu^{*}(t-\tau)e^{j2\pi\nu t}dt$$

Inserting Equations 19 and 20 produces

$$\chi(\tau,\nu)=\int_{-\infty}^{+\infty}\sum_{n=0}^{N-1}\mathrm{rect}\left[\frac{t-nT}{t_p}\right]e^{j[2\pi(b_n f_o+nf_r)(t-nT)+\varnothing_n]}$$

$$\times\sum_{m=0}^{N-1}\mathrm{rect}\left[\frac{t-\tau-mT}{t_p}\right]e^{j[2\pi(b_m f_o+mf_r)(t-\tau-mT)+\varnothing_m]}e^{j2\pi\nu t}dt. \quad (21)$$

With a change of variable of $x= t-nT$, and then letting $x = t$, Eq. 21 becomes

$$\chi(\tau,\nu)=\int_{-\infty}^{+\infty}\sum_{n=0}^{N-1}\mathrm{rect}\left[\frac{t}{t_p}\right]e^{j[2\pi(b_n f_o+nf_r)t+\varnothing_n]}\sum_{m=0}^{N-1}\mathrm{rect}\left[\frac{t-\tau-(m-n)T}{t_p}\right]$$

$$\times\ e^{-j2\pi(b_m f_o+mf_r)t}e^{-j2\pi(b_m f_o+mf_r)(n-m)T}e^{j2\pi(b_m f_o+mf_r)\tau}$$

$$\times\ e^{-j\varnothing_m}e^{j2\pi\nu t}e^{j2\pi\nu n\tau}dt. \quad (22)$$

After additional grouping of terms, Equation 22 becomes

$$\chi(\tau,\nu)=\sum_{n=0}^{N-1}\sum_{m=0}^{N-1} e^{j2\pi b_m f_o[(m-n)T+\tau]}\, e^{j2\pi m f_r[(m-n)T+\tau]}\, e^{j(\phi_n-\phi_m)}\, e^{j2\pi\nu nt}$$

$$\times \int_{-\infty}^{+\infty} \mathrm{rect}\left[\frac{t}{t_p}\right] \mathrm{rect}\left[\frac{t-\tau-(m-n)T}{t_p}\right] e^{j2\pi[\nu-(b_m-b_n)f_o-(m-n)f_r]t}\, dt \qquad (23)$$

In Equation 23, those terms grouped with the integral themselves define an ambiguity function of a single pulse, delayed by $(m-n)T$ and phase shifted by $(b_m-b_n)f_o+(m-n)f_r$. Therefore, the ambiguity function for the pulse train can be expressed as:

$$\chi(\tau,\nu)=\sum_{n=0}^{N-1}\sum_{m=0}^{N-1} e^{j2\pi b_m f_o[(m-n)T+\tau]}\, e^{j2\pi m f_r[(m-n)T+\tau]}\, e^{j(\phi_n-\phi_m)}\, e^{j2\pi\nu nt}$$

$$\times \quad \chi_c\left[\tau-(n-m)T,\nu-(b_m-b_n)f_o-(m-n)f_r\right] \qquad (24)$$

where $\chi_c(\tau,\nu)$ is the ambiguity function for a single pulse. If the initial phases and frequency drifts are zero, Equation 24 reduces to

$$\chi(\tau,\nu)=\sum_{n=0}^{N-1}\sum_{m=0}^{N-1} e^{j2\pi m f_r[(m-n)T+\tau]}\, e^{j2\pi\nu t}\, \chi_c\left[\tau-(n-m)T,\nu-(m-n)f_r\right] \qquad (25)$$

Equation 25 matches the expression given by Rihaczeck for interpulse frequency shift coding of coherent pulse trains, except that in Eq. 25 the randomness in the frequency shifts has been replaced by linear steps (Ref 9:316). The plot of Eq. 25 will form the basis for determining the effects of non-coherence to follow in later sections, and thereby warrants additional discussion.

In the description of Class C waveforms, the double summation in Eq. 15 was expanded to show the sampling effect of the exponential on each p surface. A similiar expansion of Eq. 25 would reveal a similar, but more complicated sampling effect caused by the two exponentials. In addition, the linear frequency stepping would have a shearing effect similiar to that of the linear FM pulse seen in Figure 7. Also, the additional doppler term in the ambiguity funtion for the component pulse would cause a translation of that p surface from the delay axis. The overall effects on the gross structure of the ambiguity function are depicted in Figure 10.

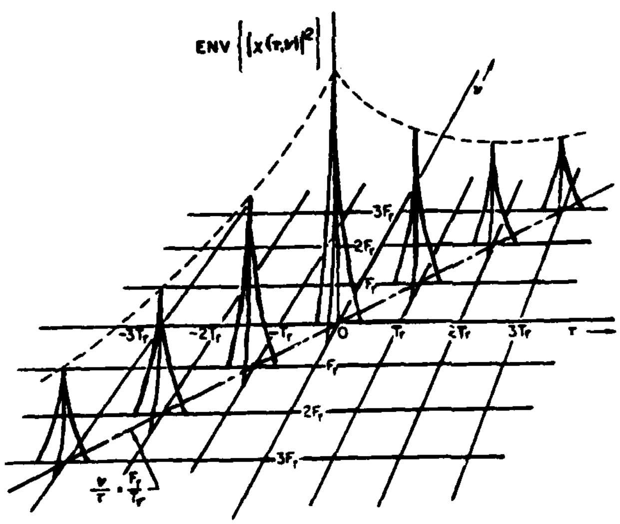


Figure 10. Gross Structure of the Ambiguity Surface for Linear Frequency Shift Coding (Ref 11:319).

Resolution Potential and the Ambiguity Function

According to Rihaczek, in a matched filter system, there is no attempt to resolve targets separated by less than the half power width of the receiver response (Ref 11:90). Accordingly, the nominal range and doppler resolution potential of a waveform in a radar using a matched filter receiver may be determined by finding the half power points of the central response lobe. A qualitative estimate of the resolution capabilities can be made from a plot of the ambiguity function since it represents the matched filter response as a function of range and doppler.

This study uses the half power point approach in determining the effects of random phases and carrier drifts on the resolution capabilities of the linear step frequency waveform. However, the magnitude of the ambiguity function, rather than the magnitude squared, is plotted and analyzed. In effect the -6dB power points are found instead of the -3dB points, but for determining the effects of random phases and drifts, the difference is unimportant.

III. Software Development

Overview

In this chapter the software required to calculate, evaluate, and plot the ambiguity function for the linear step frequency waveform is described. The software is written in Fortran for the Data General Eclipse S/250 computer in the AFIT Signal Processing Laboratory.

The starting point of the programming effort was an analysis of Capt. John Reed's program listings developed to calculate and plot ambiguity functions for a number of waveforms (Ref 9: Appendix D). The techniques used by Reed form much of the foundation for this present work; indeed, some code was duplicated. However, Reed's technique to represent a pulse train proved unworkable for the linear step frequency waveform, thus requiring virtually an entirely new set of programs to be written. In addition, the discrete Fourier transform routine used by Reed is no longer compatible with the present Eclipse operating system.

The software developed implements Eq. 23. The central focus of this study is to determine the effects on the ambiguity function of random drift in the carrier frequency (variables b_m and b_n), and random initial phases (variables $\varnothing_n$ and $\varnothing_m$). The other variables in Eq. 23 are pulse-width (t_p), pulse repetition time (T), carrier frequency (f_o), linear frequncy step size (f_r), and the number of pulses in the waveform (N). The function variables are delay (τ) and doppler frequency shift (ν). The selectable range of these variables is presented in Table II.

The software written for this study is composed of three main programs and 12 subroutines. Program CONPULS computes the ambiguity function as given by Eq. 23. Program GRAPH reformats the computed data for plotting by an existing Elcipse plotting program PLTTRNS, and optionally tabulates the data and significant variables of the waveform. Program GRAPH2 tabulates magnitude data computed by CONPULS also, but does not round the data to integer values as does GRAPH, thereby allowing determination of the half maximum magnitude value required to determine the doppler resolution. The following sections describe the specifics of assumptions, limitations, and techniques of the software.

<u>Calculation of $\chi(\tau,\nu)$ by Program CONPULS</u>

The ambiguity function for the linear step frequency waveform given by Eq. 23 is composed of the superposition of 2N ambiguity functions of a single component pulse. Recall that the integral in Eq. 23 was defined to be the ambiguity function for the component pulse. That relationship is given by

$$\chi_c(\tau,\nu) = \int_{-\infty}^{+\infty} \mathrm{rect}\left[\frac{t}{t_p}\right] \mathrm{rect}\left[\frac{t-\tau-(m-n)T}{t_p}\right] e^{j2\pi[\nu-(b_m-b_n)f_o-(m-n)f_r]t}\, dt \quad (26)$$

TABLE II. Program Selectable Variable Ranges

Variable(s)	Selectable Ranges
b_m, b_n	Pseudo random uniform distribution over selectable percentage 0.0 to 1.0 of carrier frequency, fo.
$\varnothing_m, \varnothing_n$	Pseudo random distribution over 2 ranges: 20 or 180 degrees.
t_p	Rectangular pulses: 4.9 to 0.000001 seconds.
T	2 or 4 times tp: 9.8 to 0.000002 seconds.
f_r	Frequency step size: 0.0 or $(1/t_p)$ Hz.
f_o	Nominally set to $100*(1/t_p)$ Hz. Manual range: 100 to 1 GHz.
N	2, 4, or 8 rectangular pulses.
τ	Determined by T and N; maximum value equal to +/- N*T
ν	64 equally spaced values between upper and lower limits. Limits nominally set to +/- $2*N-1*(1/t_p)$ Hz. No manual limits set.

The basic approach in evaluating Eq. 23 is to evaluate Eq. 26 for one value of m and n and specific values of τ and ν, then multiply the result by the leading phase terms and add the results to the preceding results (if any). The variables τ and ν are varied appropriately over the range of interest. In effect, for all values of m and n, an ambiguity surface is created and then added, point by point, to the other

surfaces created for all m and n. The end result is the superposition of 2N ambiguity surfaces, where N is the number pulses in the waveform.

In a simplified representation, Eq 26 becomes

$$\chi_c(\tau,\nu) = \int_{-\infty}^{+\infty} \mu(t)\ \mu^*(t-\tau)\ e^{j\omega t}\ dt \tag{27}$$

Equation 27 describes the cross correlation of $\mu(t)e^{j\omega t}$ and $\mu(t)$. However, it can be shown that $\chi(\tau,\nu)$ may be written as a convolution (Ref 5:172-173) given by

$$\chi_c(\tau,\nu) = \mu(\tau)e^{j\omega\tau} \otimes \mu^*(-\tau) \tag{28}$$

where $\otimes$ indicates the convolution operator, and * indicates the complex conjugate. In this form $\chi_c(\tau,\nu)$ may be evaluated with a digital computer since

$$\mu(\tau)e^{j\omega\tau} \otimes u^*(-\tau) = \mathfrak{F}^{-1}\left[U(f-f_0)U(-f)\right] \tag{29}$$

In words Eq. 29 states that $\chi(\tau,\nu)$ is equivalent to the inverse Fourier transform of the product of the transforms of $\mu(t)e^{j\omega t}$ and $\mu(-t)$. In general, the operations indicated by Eq. 29 can be accomplished by first representing each complex envelope by a finite sequence of length N samples, and then adding N-1 zero-valued samples. The discrete Fourier transforms of both sequences are computed, the results of which are multiplied together, point by point. The linear convolution is completed by taking the inverse discrete Fourier transform of the product.

In this study, the discrete Fourier transform operations are performed by subroutine FOUREA. This subroutine was extracted from Reference 8:1-5 and adapted for use on the Eclipse. The routine computes a one dimensional fast Fourier transform (FFT) by using a radix 2 decimation in time algorithm on the real and imaginary parts of the complex data input. The input data is passed via a 256 element complex array. As originally adapted for the Eclipse, the size of the input array could be any integer power of two between 1 and 15. The size 256 represents a practical compromise necessitated by the limited Eclipse memory and the inherent ineffficency of the subroutine.

The approach in representing the rectangular pulses in complex arrays is as follows. Two 256 element complex arrays are created to represent $\mu(t)e^{j\omega t}$ and $\mu(-t)$. In each, array element 128 represents time (τ) zero. The pulse is represented by setting the real part of a number of array elements equal to one. The number of elements used to represent $\mu(t)$ and $\mu(-t)$ is a function of the ratio of pulse repetition interval (T) to the pulsewidth (t_p), the number of pulses in the waveform (N), and the array size. For example, the correlation of a rectangular pulse train where N two is and T/t_p equals four would result in triangular shapes spaced as shown in Figure 11.

The triangular shapes occupy a width of $2*[(N-1)(T/t_p)+1]$ pulsewidths, or 10 t_p in this case. With an array size of 256, this would allow t_p to be represented by 256/10=25.6 pulses. An non-integer number is not satisfactory. However, if the total number of pulsewidths in the array is required to be a power of two, then the number of array elements

representing the pulses will be an even integer. For instance, in the example above, if N and (T/t_p) are required to be integer powers of two, then $2*N*(T/t_p)$ is also an integer power of two (16), and greater than the total number of pulsewidths occupied by the correlation result. Then the number of elements to represent a pulse (NUM) will be an even integer given by

$$NUM = 256/[2*N*(T/t_p)] \tag{30}$$

The key is to restrict N and (T/t_p) to integer powers of two. As shown in Table II, these restrictions are implemented. Additional details are given in the listing of subroutine INDAT.

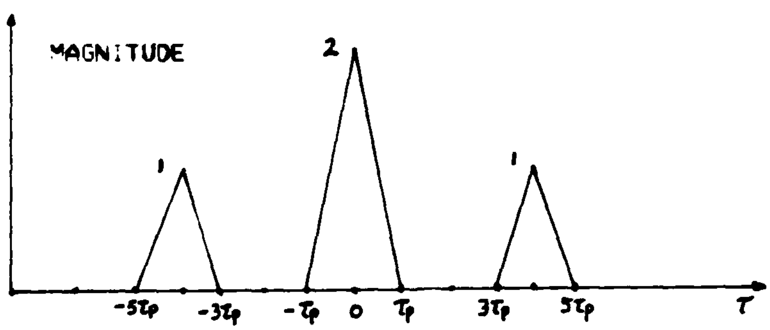


Figure 11. Correlation of 2 Pulse Rectangular Envelope Pulsetrain.

With the number of samples per pulse (NUM) and the ratio T/t_p (RATIO) determined, the task of representing a pulse in the arrays is

simplified. Array elements 128 to 128+NUM are set equal to 1.0 in one array to represent $\mu(t)$, while in the second elements (128-(m-n)*RATIO) to (128-(m-n)*RATIO+NUM) represent $\mu(-t)$. It is seen then that $\mu(-t)$ will shift position as m and n vary. All other elements of both arrays are set to zero.

The exponential term in Eq. 26 multiplies the array elements representing $\mu(t)$. Since the exponential is a function of time, a method to represent time must be chosen. If the first non-zero element, element 128, represents τ equal zero, then for each successive element, the value of tau is given by

$$\tau = (t_p/\mathrm{NUM})*n \tag{30}$$

where t_p is the pulsewidth in seconds, NUM is the number of elements representing the pulse, and n *is* the nth element from 128. Each array element representing $\mu(t)$ is then multiplied by the appropriate value of $e^{j\omega t}$. The actual expression for the exponential is that within the integral in Eq. 23. The loading of the arrays representing $\mu(t)e^{j\omega t}$ and $\mu(-t)$ is accomplished by subroutine LDARYS.

Each array is then passed to subroutine FOUREA, where the discrete Fourier transform of each is computed. The results of the transformations, which reside in the original arrays, are then multiplied together element by element, with the product replacing the contents of one array. That array is then inverse transformed to complete the convolution.

The results, contained in the X array, must also be multiplied by

the phase terms preceding the integral in Eq. 23. The phase terms which are not a function of τ simply multiply each element in the X array. Those terms which are a function of τ are handled similarly to the exponential term multiplying $\mu(t)$ within the integral. See the listing of program CONPULS, do-loop 80 for the specific details.

At this point in the computation, the X array contains 256 complex values for 256 points of τ for one specific value of doppler shift (ν). These results must be added point by point to those for the same values of τ and ν but different m and n. But not every value can be retained because the plotting program PLTTRNS can plot an array no larger than 64 x 64. Therefore, only every fourth value in the X array is added to previous results. Also, since the effects of the leading phase terms are important to preserve, the real and imaginary parts of each value, not the magnitude, must be added to the corresponding real and imaginary parts from previous results.

Before the addition takes place however, each real and imaginary value is normalized by multiplication with the variable SCALE. Recall that in Eq. 8 it was stated that the maximum value of $\chi(\tau,\nu)$ occurs at $\tau = \nu = 0$ and is equal to one (assuming a normalized complex envelope). In performing the convolution of two pulses via the discrete Fourier transform, the maximum value obtained is equal to the number of pulses times the number of elements representing each pulse times the amplitude of the pulse. Therefore, SCALE is set equal to the reciprocal of the product of the number of pulses and the number of elements representing each pulse. In this manner, the normalization of the ambiguity surface

is accomplished as is customary. The results are stored in two 64 x 64 real arrays, RANS and IANS.

The algorithm to evaluate the integral in Eq. 23 is repeated 64 times over the doppler range of interest for each value of m and n. The method of representing $\mu(t)$ and $\mu(-t)$ in arrays as a function of time eliminates the need to explictly vary the delay parameter τ as is the doppler parameter ν. The maximum value of τ represented in this manner is $128 * t_p$/NUM. The maximum represents the total duration of the pulse train. The corresponding range may be derived from Eq. 7.

After all 64 values of ν have been evaluated for the first values of m and n (m=n=0), arrays RANS and IANS contain the real and imaginary parts of the ambiguity function for one value of (m-n). Plotted by themselves, these values would depict the ambiguity function of a single rectangular pulse as shown in Figure 2. However, do-loops 30 and 35 in CONPULS sequence the process through all values of m and n, until RANS and IANS contain the complete answer. At this point, the magnitude of each pair of real and imaginary values is computed one row at a time, and written to file ANS. This data file will be read and reformatted by program GRAPH in a form suitable for program PLTTRNS. Execution time for the program varies as the number of pulses squared. For a 2 pulse train, four ambiguity surfaces are computed, each composed of 64 cuts. Each cut takes one second to be computed. Then for a two pulse train, the execution time is 1 second multiplied by 64 multiplied by 2^2, or 4.25 minutes. The execution times for four and eight pulse trains are 17.1 minutes and 1.1 hours respectively.

The last subroutine in CONPULS is SAVDAT. Its purpose is to save the parameters listed in Table III for use by GRAPH.

TABLE III. Parameters saved by SAVDAT

Parameter	Source Subroutine	Significance
Time	INDAT	Represents the time between samples of each pulse: TIME = PW/NUM.
WS	INDAT	Starting value of doppler shift. Nominally -2*(NP-1)*1/PW.
WF	INDAT	Final value of doppler shift: WS and WF may be set manually.
WIN	INDAT	Doppler interval = (WS+WF)/64.
NP	INDAT	Number of pulses in the waveform: 2, 4, or 8
PW	INDAT	Pulsewidth: 4.9 to 0.000001 seconds.
PRT	INDAT	Pulse repetition interval: 2 or 4 time the pulsewidth:
RANGE	PERRORS	Initial phase range: 20 or 180 deg.
TESTP	PERRORS	Control variable to select initial phases: 1= yes, 0= no.
TESTF	FERRORS	Control variable to select carrier drifts: 1= yes, 0=no.
FO	FERRORS	Carrier frequency: Nominally set to 100*(1/PW). Set to 0.0 if TESTF= 0.
PERCENT	FERRORS	Maximum percentage of FO assigned to random drifts in FO.
PE1,PE2	PERRORS	8 element array containing random initial phases assigned to pulses.
FE1,FE2	FERRORS	8 element array containing random carrier drifts assigned to pulses.

Generation of Initial Phases and Carrier Drifts

The random initial phases and random carrier drifts indicated within Eq. 23 are generated by subroutines PERRORS and FERRORS respectively. Both routines are similar and therefore described together. Each routine is called from within CONPULS, immediately displays a message describing its purpose, and presents the option to include random phases or carrier drifts. If the option is declined, a number of elements equal to the number of pulses in arrays PE1 and PE2 or FE1 and FE2 are set to zero. These arrays hold the initial phases and carrier drifts.

If the option to include initial phases in the calculations is selected, PERRORS presents two ranges of phases from which to choose: 20 and 180 degrees. In FERRORS, first the option to automatically set the carrier frequency FO to 100*(1/PW) is presented, and then to set the maximum percentage of carrier drift to 1.0. Both options allow manual input if the nominal value is declined. The values for the phase ranges, FO and maximum percentage drift (PERCNT) represent reasonable values for those variables.

The heart of both routines is a modified algorithm given by Dorn and Greenberg to generate pseudo random integers (Ref 4:481). The algorithm relies on a process of comparing an initial seed to two integers, L and M, which are primitive roots of two. The algorithm produces a pseudo random number between 2 and L+M-2. The cycle length is the least common multiple of L-1 and M-1.

The algorithm in PERRORS uses primitive roots 179 and 181 to generate random numbers between 2 and 358, while FERRORS uses 491 and 509 to generate numbers between 2 and 998. The primitive roots were selected from a table as opposed to computed (Ref 1:113).

In both routines the random numbers generated are multipied by factors to keep them within the ranges initially selected prior to storage in the arrays PE1, PE2, FE1 and FE2. Both routines include an option to display the phases or drifts stored prior to exiting the routines. If the values are clustered too closely or otherwise unsatisfactory, the option to restart each routine and to input a different seed is offered.

<u>Subroutine TST1</u>

The discussion of program CONPULS concludes with a description of subroutine TST1. This subroutine allows a printout of all 256 values of the ambiguity function corresponding to a specific value of doppler shift. In the jargon of radar resolution, this process is called taking a "cut" or row of the ambiguity surface. Since each value has a corresponding value of delay (τ) associated with it, this routine is used to examine the effects of random initial phases and random carrier drifts as will be explained in Chapter IV.

A complete program listing for CONPULS and its subroutines is contained in Appendix A. The listings have been liberally commented to facilitate understanding and modification.

Formatting and Tabulating Data with Program GRAPH

Program GRAPH has two purposes. The first is to reformat the data file ANS created by CONPULS. The file contains the magnitude values of $\chi(\tau,\nu)$, but in a format not acceptable by the plotting program PLTTRNS. The second purpose is to optionally tabulate the reformatted data for a specific row or column in order to examine its symmetry, or to determine its maximum and -3dB values. The tabulation option includes printing the variables in Table III in order to produce a record of the waveform parameters creating the data. What follows is an amplification of these basic purposes.

Recall that program CONPULS wrote the 64 x 64 array of computed magnitude values into file ANS. Program GRAPH begins by reading this file into the 64 x 64 real array ANS, then continues by reading the variables of Table III from file PAMS. The program then scales each array element in array ANS to an integer value between 1 and 200, and stores the new values in the 64 x 64 integer array GRAPH. Scaling to integer values is required by PLTTRNS and smoothes the data as well. Once the scaling is complete, array GRAPH is read one row at a time into the real part of the complex array STORE. The imaginary part of STORE is set to zero. This action is again required by PLTTRNS since the input to it must be in a complex, integer array. See the listing for GRAPH through statement number 115 for details of the reformatting process.

Subroutine TABULATE is called if the option to tabulate row or column data is selected from within GRAPH. TABULATE is similar to TST1 in program GRAPH. However, this routine allows multiple iterations in

order to create tables for more than one row or column. It also presents the option to produce hardcopy as opposed to automatically doing so. The maximum value and the nearest -3dB value within the selected row or column is found as are the corresponding values of delay or doppler shift. Then all 64 magnitude values and corresponding delay or doppler values are printed in columns. Finally, subroutine VARS is called to print the variables of Table III.

The program listing for GRAPH and its subroutines is contained in Appendix B.

Program Graph2

The purpose of subroutine GRAPH2 is to tabulate the datafile ANS similarily to GRAPH, but without converting the data to integers. By leaving the data in real form, a more accurate calculation of delay and doppler resolution may be obtained. The program listing for GRAPH2 is contained in Appendix C.

Program PLTTRNS

Many of the steps in programs CONPULS and GRAPH were necessitated by the requirements of PLTTRNS. Since neither the source code nor formal instructions for running the program are available, this section is included to provide continuity in moving from the software created for this study and PLTTRNS. What follows comes primarily from Reed's work (Ref 9:49-51,100) and is augmented by this author's experience.

PLTTRNS was written by R.W. Schafer in July 1978 at Georgia Institute of Technology. The routine is evoked from the RDOS operating system prompt, R, by typing PLTTRNS <filename>/I <array size>/N, where <filename> refers to the name of the file containing the data to be plotted. The input data array is assumed to be square, may be up to 512 x 512 and no smaller than 64 x 64. The data must be in complex integer format. The program provides three plotting modes: a contour of on half the array; a three dimensional plot; and a two dimensional plot of any one of the array's 64 rows. The contour and 3-D plots can only accept a 64 x 64 input array. Plots are generated on a Tektronix 4010 compatible terminal and hardcopied with a Tektronix 4631 Hardcopy Unit.

IV. Software Validation

Constant Carrier Simulation

The first step in validating the software was to include the capability to evaluate the ambiguity function of a constant carrier pulse-train. By setting the frequency step size (f_r), the random initial phase terms ($\emptyset_i$), and the random carrier drift terms (b_i) to zero, Eq. 23 is reduced to that of the constant carrier, uniform pulse train expression of Eq. 15. The plot of this expression is well known and would serve as validation for the basic software techniques used.

In initial testing of this capability, multiplication by the leading exponential terms of Eq. 23 was bypassed. The results obtained were incorrect because the internal surfaces of the plot did not have the characteristic sin Nx/sin x breakup as seen in Figures 8 and 9. Recall that the breakup is caused by the term

$$\frac{\sin \pi\nu(N-|p|)T}{\sin \pi\nu T}$$

in Eq. 16, herein called the grating function, and results from the expansion of the double sum and the term $e^{j2\pi\nu nt}$ in Eq 15.

In plotting Eq. 16, the exponential term involving p can be ignored without effecting the grating function as it drops out when the envelope of each p surface is computed (Ref 4:186). Thus the magnitude of each p surface is computed individually and then summed with all others. But in order to examine the effects of random initial phases and carrier drifts,

the magnitude must be computed only after the summation of all p surfaces.

Therefore, the term $e^{j2\pi\nu t}$ must be included. However, inclusion of the term as given also produced incorrect results. The maximum value of the plot was not at the origin, but was displaced along the doppler axis in the negative direction. Also, the magnitudes along the axis were not symmetric. These effects result from the dependence of the exponential on the value of n. For a pulse train of two pulses, two pairs of real and imaginary data for each value of τ and ν combine to form the central p surface (m-n=0). Clearly, the pairs will not be equal since the exponential would multiply one pair and not the other. The result is the shift of the maximum value and the asymmetry described above.

Changing the exponential to $e^{j2\pi(n+1)\nu t}$ resulted in obtaining the expected results. Figure 12 is the plot produced for a train of four pulses, where the pulsewidth is 0.001 seconds, the pulse repetition time (PRT) is 0.002 seconds, and the total doppler spread is 1000 Hz. The figure shows all the characteristics expected of a plot of a constant carrier pulsestrain. There are N-1=3 p surfaces to each side of the doppler axis. The central peak is greatest for p equal zero, and is smallest for |p| equal 3. The grating function effect is visible in each p surface with the greatest breakup for p equal zero and none for |p| equal three. The doppler width of the central peak in each p surface increases for increasing p. The total number of peaks in each p surface is equal to the PRT divided by the pulsewidth.

Tables 4 and 5 show the symmetry expected for ν equal zero and τ equal zero respectively. With some interpretation, these tables also validate the amplitudes expected. Recall that the maximum value of the

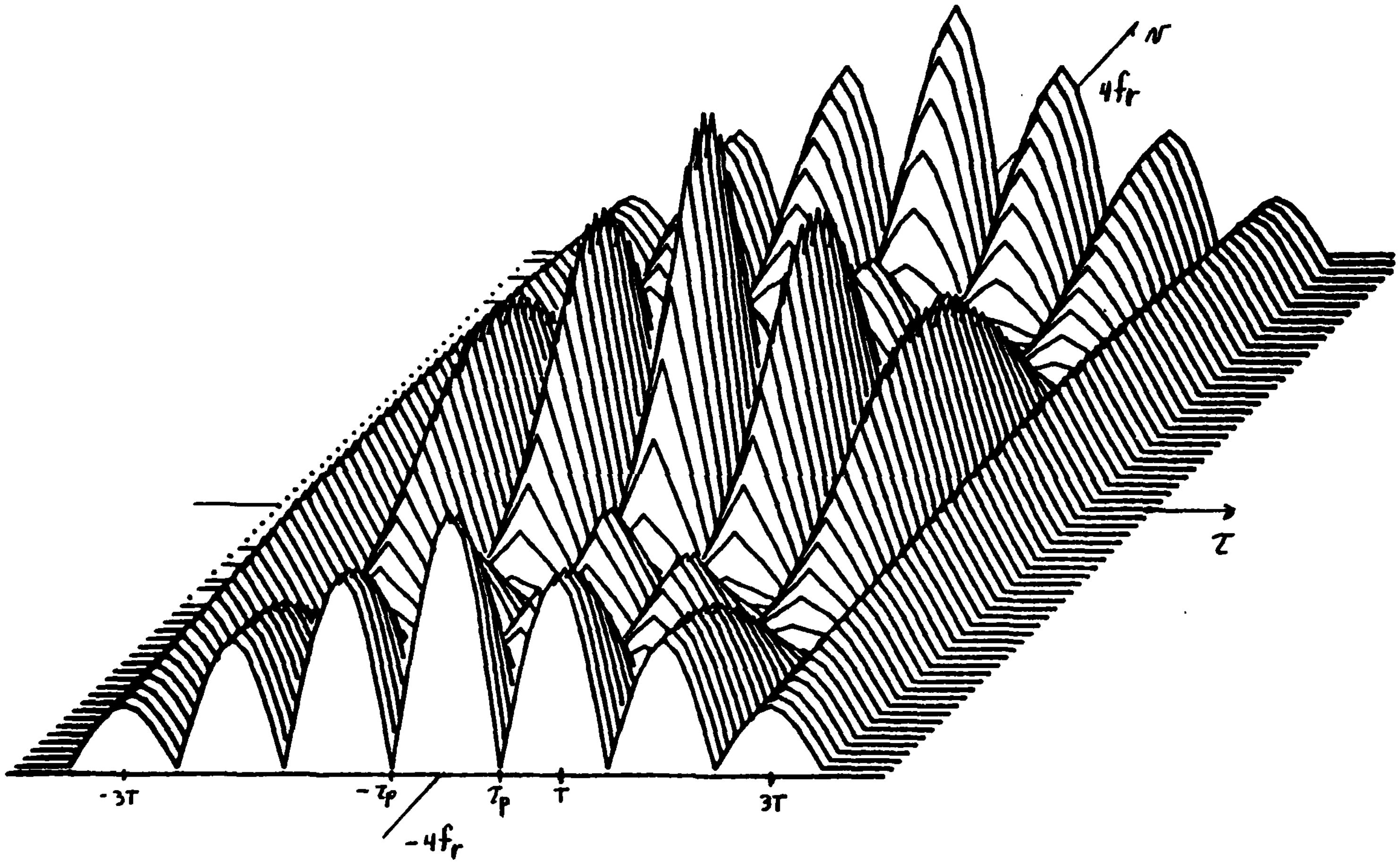


Figure 12. $|\chi(\tau,\nu)|$ for a Constant Carrier Pulsetrain, N=4, T/t_p =2.

TABLE IV. Magnitude Data for $\chi(\tau,0)$ of a Constant Carrier Pulsetrain, N=4.

ROW #= 32 (DOPPLER SHIFT= -.000)

MAXIMUM VALUE = 199 AT TAU= .000000 (SAMPLE # 32)
-3dB VALUE = 99 AT TAU= .000500 (SAMPLE #34)

TAU	MAGNITUDE	TAU	MAGNITUDE
-.007749997	0	.000250000	149
-.007499997	0	.000500000	99
-.007249996	0	.000750000	49
-.006999999	0	.001000000	0
-.006749999	12	.001250000	37
-.006499998	24	.001500000	74
-.006249998	37	.001750000	112
-.005999997	49	.002000000	149
-.005749997	37	.002250000	112
-.005499996	24	.002500000	74
-.005249999	12	.002750000	37
-.004999999	0	.003000000	0
-.004749998	24	.003250000	25
-.004499998	49	.003500000	49
-.004249997	74	.003750000	74
-.003999997	99	.003999997	99
-.003750000	74	.004249997	75
-.003500000	49	.004499998	50
-.003250000	24	.004749998	25
-.003000000	0	.004999999	0
-.002750000	37	.005249999	12
-.002500000	74	.005499996	24
-.002250000	112	.005749997	37
-.002000000	149	.005999997	49
-.001750000	112	.006249998	37
-.001500000	74	.006499998	25
-.001250000	37	.006749999	12
-.001000000	0	.006999999	0
-.000750000	49	.007249996	0
-.000500000	99	.007499997	0
-.000250000	149	.007749997	0
.000000000	199	.007999998	0

WAVEFORM PARAMETERS: # OF PULSES= 4 PULSEWIDTH= .001000
PRT= .002000 # OF SAMPLES= 16 SPACING= .000062500

DOPPLER PARAMETERS: 1ST FREQ= -3141.6
LAST FREQ= 3141.6 FREQ INCREMENTS= 98.17499

INITIAL PHASES: OPTION DECLINED (ALL SET TO ZERO)

CARRIER DRIFT: OPTION DECLINED (ALL SET TO ZERO)

TABLE V. *Magnitude Data for $\chi(0,\nu)$ of a Constant Carrier Pulsetrain, N=4.*

COLUMN #32 (TAU = .000000)

MAX VALUE = 199 AT W= .000 (SAMPLE #32)
-3dB VALUE= 97 AT W= 490.873 (SAMPLE #37)

DOPPLER SHIFT	MAGNITUDE	DOPPLER SHIFT	MAGNITUDE
-3043.418	128	98.174	195
-2945.243	122	196.348	180
-2847.069	110	294.523	158
-2748.894	93	392.698	129
-2650.719	71	490.873	97
-2552.544	47	589.047	62
-2454.370	23	687.222	29
-2356.195	0	785.397	0
-2258.020	19	883.572	23
-2159.845	34	981.743	40
-2061.671	43	1079.919	49
-1963.496	45	1178.095	51
-1865.321	41	1276.271	45
-1767.146	31	1374.442	33
-1668.972	17	1472.618	17
-1570.797	0	1570.794	0
-1472.622	17	1668.966	17
-1374.448	33	1767.142	31
-1276.273	45	1865.317	41
-1178.098	51	1963.493	45
-1079.923	49	2061.665	43
-981.749	40	2159.841	34
-883.574	23	2258.017	19
-785.399	0	2356.192	0
-687.224	29	2454.364	23
-589.050	62	2552.540	47
-490.875	97	2650.716	71
-392.700	129	2748.892	93
-294.525	158	2847.063	110
-196.351	180	2945.239	122
-98.176	195	3043.415	128
-.001	199	3141.591	127

WAVEFORM PARAMETERS: # OF PULSES= 4 PULSEWIDTH= 001000
PRT= .002000 # OF SAMPLES= 16 SPACING= .000062500

DOPPLER PARAMETERS: 1ST FREQ= -3141.6
LAST FREQ= 3141.6 FREQ INCREMENTS= 98.17474

INITIAL PHASES: OPTION DECLINED (ALL SET TO ZERO)

CARRIER DRIFT: OPTION DECLINED (ALL SET TO ZERO)

correlation of the unit amplitude pulsetrain is the product of the number of pulses and the number of samples per pulse. Recall also that each value is first scaled between 0.0 and 1.0, with 1.0 corresponding to the maximum, and secondly scaled between 0 and 200 for plotting purposes. For instance, a maximum value of 64 is first scaled to 1.0 and then to 200. In Table 4, the number of samples representing each pulse is shown as 16 and the number of pulses as 4. Their product is 64. The maximum value at τ equal zero then should have been 200, not 199 as shown. The appparent discrepancy results from computer truncation of a value of 0.999 to 0.000 when the value is assigned to an integer variable as occurs in program GRAPH.

The maximum value of the central peak in each other p surface is (N-1) times the number of samples per pulse. These peak values are 149, 99, and 49 as would be predicted by analysis identical to that for computing the value at the origin as above. The p not equal to zero peaks are seen to occur at τ =pT as required by theory. Figure 13 is a plot of the values of Table 4 and graphically shows the symmetry expected and the characteristic triangular shapes associated with correlation of rectangular pulses. Figure 14 is a five level contour plot of Figure 12 and presents another view of the characteristics of the waveform.

Figure 15 plots $\chi(\tau,\nu)$ for a two pulse waveform, a pulsewidth of 0.001 seconds, and a PRT of 0.004 seconds. Clear areas around the central response required by theory are clearly visable. Table 6 lists all 256 values of τ of $\chi(\tau,0)$. This table format, created by subroutine TST, shows the magnitudes prior to scaling and offers complete examination of delay data. It shows that for a unit amplitude pulsetrain where

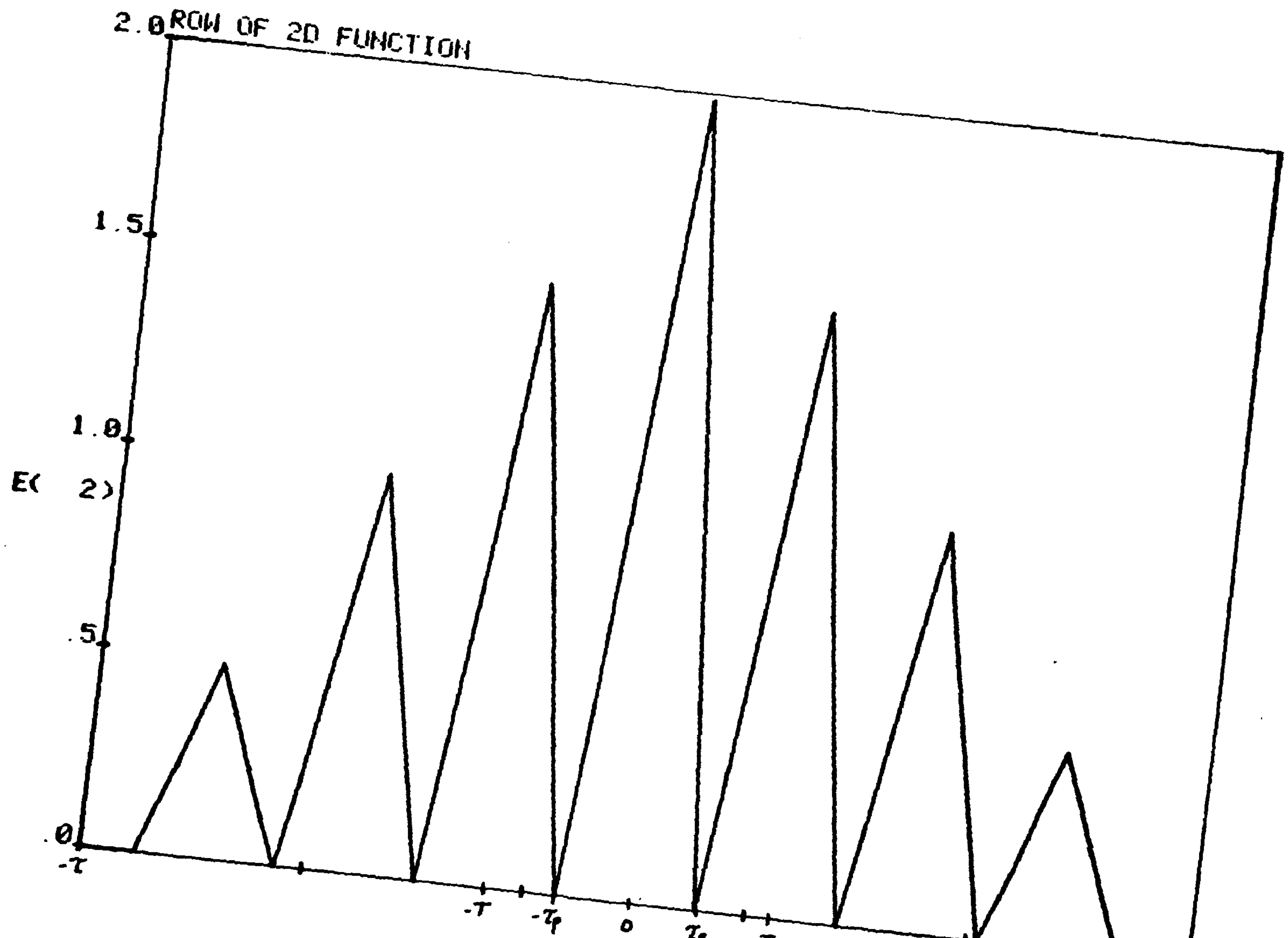


Figure 13. $|X(\tau,0)|$ for a Constant Carrier Pulsetrain, N=4, T/t_p =2.

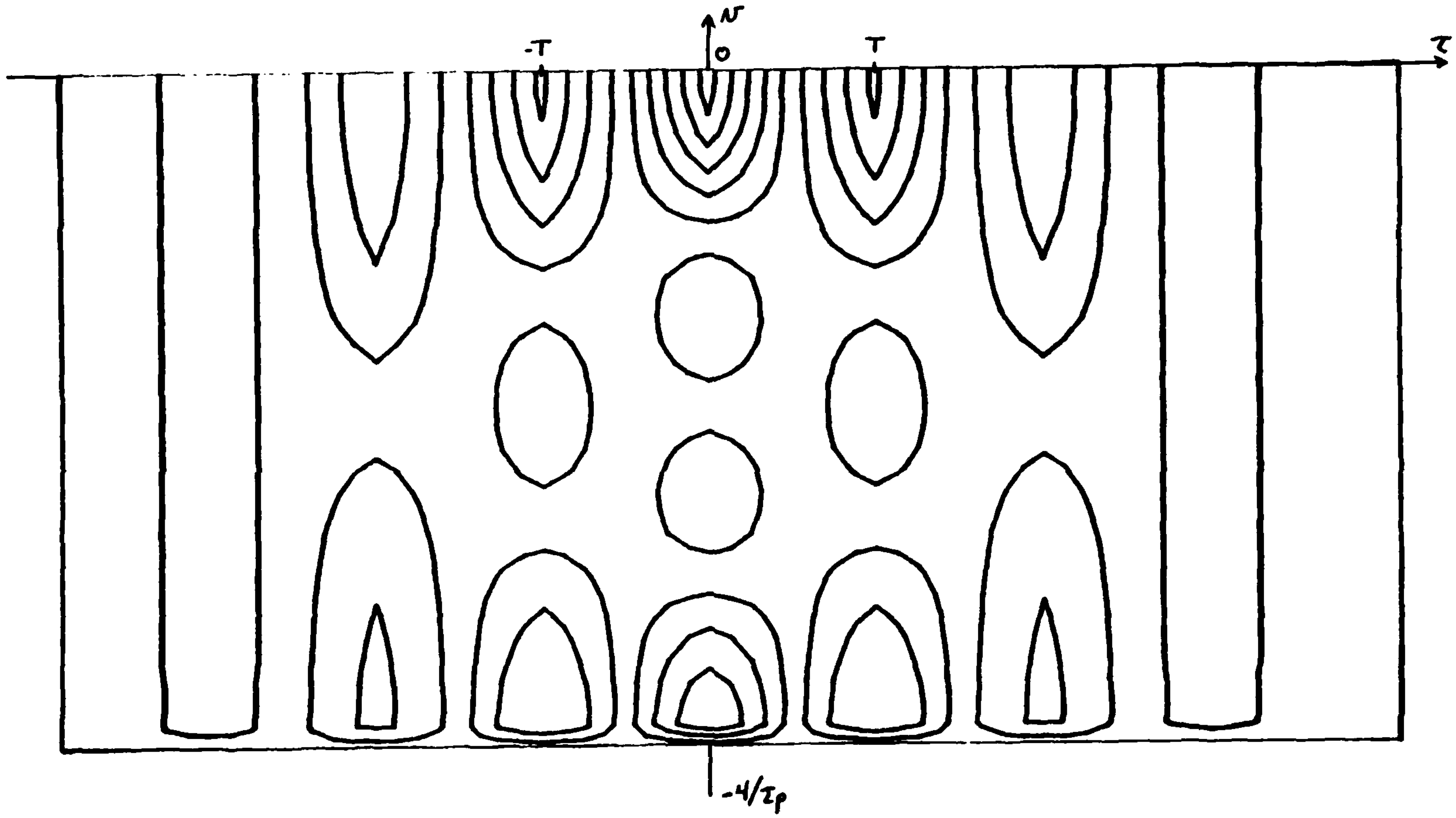


Figure 14. 5 Level Contour of $|\chi(\tau,\nu)|$ for a Constant Carrier Pulsetrain, N=4, T/t_p =2.

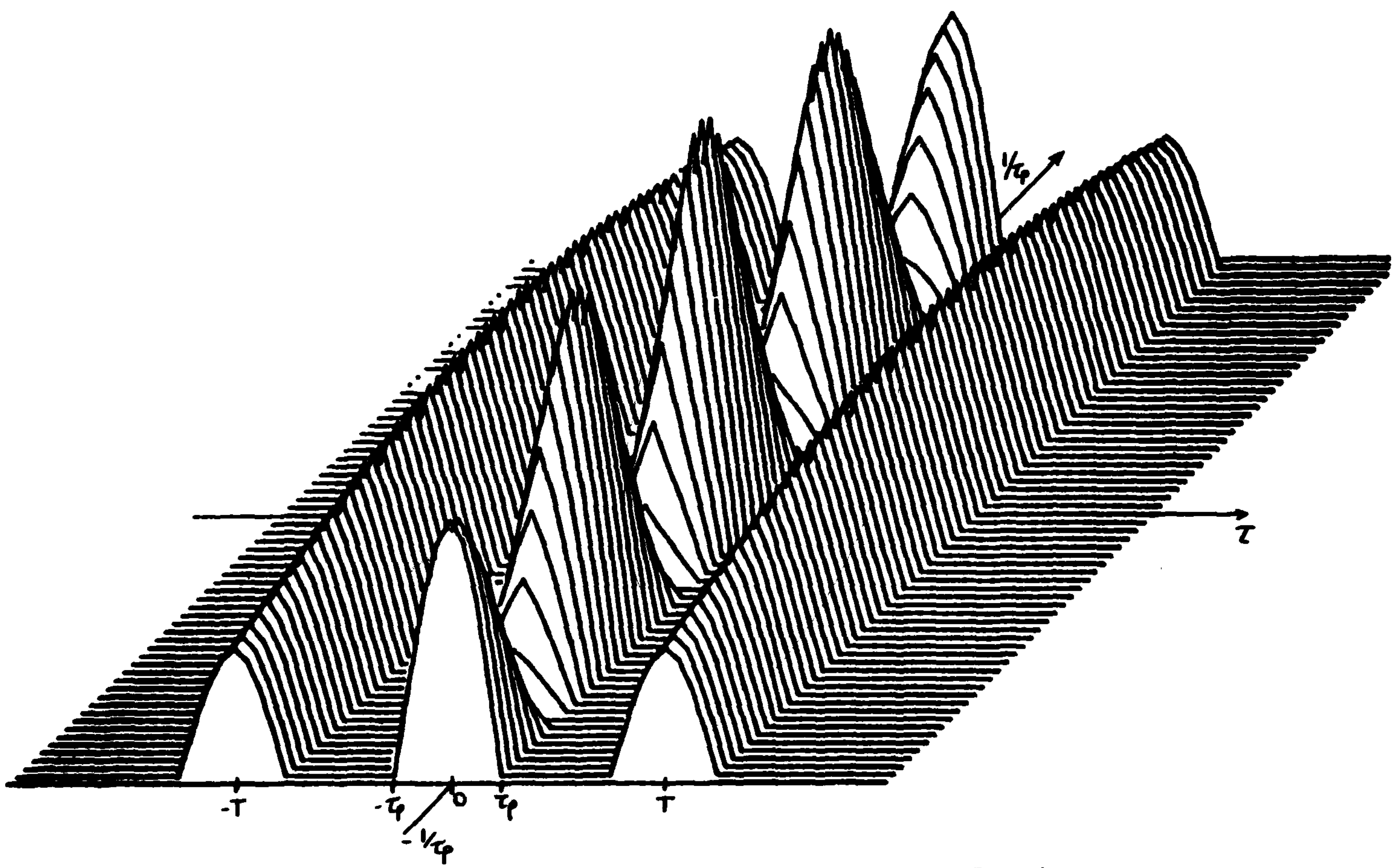


Figure 15. $|\chi(\tau,\nu)|$ for a Constant Carrier Pulsetrain, N=2, T/t_p =4.

TABLE VI. Magnitude Data for $\chi(\tau,0)$ of a Constant Carrier Pulsetrain, N=2.

****** DATA FOR ROW 32 *****

MAXIMUM VALUE= 32.000 AT TAU= .00000 (SAMPLE 128)
-3DB VALUE= 16.000 AT TAU= .00050 (SAMPLE 136)

1-32	33-64	65-96	97-128	129-160	161-192	193-224	225-256
.000	.000	15.000	.000	30.000	.000	15.000	.000
.000	.000	14.000	.000	28.000	.000	14.000	.000
.000	.000	13.000	.000	26.000	.000	13.000	.000
.000	.000	12.000	.000	24.000	.000	12.000	.000
.000	.000	11.000	.000	22.000	.000	11.000	.000
.000	.000	10.000	.000	20.000	.000	10.000	.000
.000	.000	9.000	.000	18.000	.000	9.000	.000
.000	.000	8.000	.000	16.000	.000	8.000	.000
.000	.000	7.000	.000	14.000	.000	7.000	.000
.000	.000	6.000	.000	12.000	.000	6.000	.000
.000	.000	5.000	.000	10.000	.000	5.000	.000
.000	.000	4.000	.000	8.000	.000	4.000	.000
.000	.000	3.000	.000	6.000	.000	3.000	.000
.000	.000	2.000	.000	4.000	.000	2.000	.000
.000	.000	1.000	.000	2.000	.000	1.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	1.000	.000	2.000	.000	1.000	.000	.000
.000	2.000	.000	4.000	.000	2.000	.000	.000
.000	3.000	.000	6.000	.000	3.000	.000	.000
.000	4.000	.000	8.000	.000	4.000	.000	.000
.000	5.000	.000	10.000	.000	5.000	.000	.000
.000	6.000	.000	12.000	.000	6.000	.000	.000
.000	7.000	.000	14.000	.000	7.000	.000	.000
.000	8.000	.000	16.000	.000	8.000	.000	.000
.000	9.000	.000	18.000	.000	9.000	.000	.000
.000	10.000	.000	20.000	.000	10.000	.000	.000
.000	11.000	.000	22.000	.000	11.000	.000	.000
.000	12.000	.000	24.000	.000	12.000	.000	.000
.000	13.000	.000	26.000	.000	13.000	.000	.000
.000	14.000	.000	28.000	.000	14.000	.000	.000
000	15.000	000	30.000	.000	15.000	.000	.000
000	16.000	000	32.000	000	16.000	.000	000

the number of pulses (N) is two and the number of samples per pulse (NUM) is equal to 16, the maximum value is

$$N * NUM = 32.0$$

as shown at the bottom of the column labeled 97-128. Table 7 lists the magnitudes for $\chi(0,\nu)$ as well as the waveform parameters used in creating Figure 15. Examination of magnitude versus doppler shift show the peaks within the center response to occur at intervals given by

$$w = 2\pi\nu = n * 1570.8 \text{ radians/second}$$

where n is the nth peak within the response. These values satisfy the relation $\nu=k/T$ as stated in the Chapter II discussion of Class C waveforms.

<u>$\chi(\tau,\nu)$ of the Basic Linear Step Frequency Waveform</u>

This section describes the validation of the software which evaluates the basic linear step frequency waveform of Eq. 25. The program executes as before except that the option to set the frequency step size to zero is declined. The step size is then automatically set to $1/t_p$. Both options to include random initial phases and random carrier drifts are declined.

The plot of $\chi(\tau,\nu)$ for a four pulse waveform is presented in Figure 16. It agrees with the drawing of the anticipated gross structure presented in Figure 10. The plot shows three p surfaces to either side of

TABLE VII. Magnitude Data for $\chi(0,\nu)$ of a Constant Carrier Pulsetrain, N= 2.

COLUMN #32 (TAU = .000000)

MAX VALUE = 199 AT W= (SAMPLE #32)
-3dB VALUE= 75 AT W= 589.047 (SAMPLE #38)

DOPPLER SHIFT	MAGNITUDE	DOPPLER SHIFT	MAGNITUDE
-3043.418	128	98.174	196
-2945.243	125	196.348	184
-2847.069	115	294.523	165
-2748.894	101	392.698	140
-2650.719	81	490.873	110
-2552.544	57	589.047	75
-2454.370	29	687.222	38
-2356.195	0	785.397	0
-2258.020	31	883.572	37
-2159.845	62	981.743	73
-2061.671	92	1079.919	105
-1963.496	119	1178.095	133
-1865.321	143	1276.271	155
-1767.146	161	1374.442	170
-1668.972	174	1472.618	178
-1570.797	180	1570.794	180
-1472.622	178	1668.966	174
-1374.448	170	1767.142	161
-1276.273	155	1365.317	143
-1178.098	133	1963.493	119
-1079.923	105	2061.665	92
-981.749	73	2159.841	62
-883.574	37	2258.017	31
-785.399	0	2356.192	0
-687.224	38	2454.364	29
-589.050	75	2552.540	57
-490.875	110	2650.716	81
-392.700	140	2748.892	101
-294.525	165	2847.063	115
-196.351	184	2945.239	125
-98.176	196	3043.415	128
-.001	199	3141.591	127

WAVEFORM PARAMETERS: # OF PULSES= 2 PULSEWIDTH= .001000
PRT= .004000 # OF SAMPLES= 16 SPACING= .000062500

DOPPLER PARAMETERS: 1ST FREQ= -3141.6
LAST FREQ= 3141.6 FREQ INCREMENTS= 98.17474

INITIAL PHASES OPTION DECLINED (ALL SET TO ZERO)

CARRIER DRIFT OPTION DECLINED (ALL SET TO ZERO)

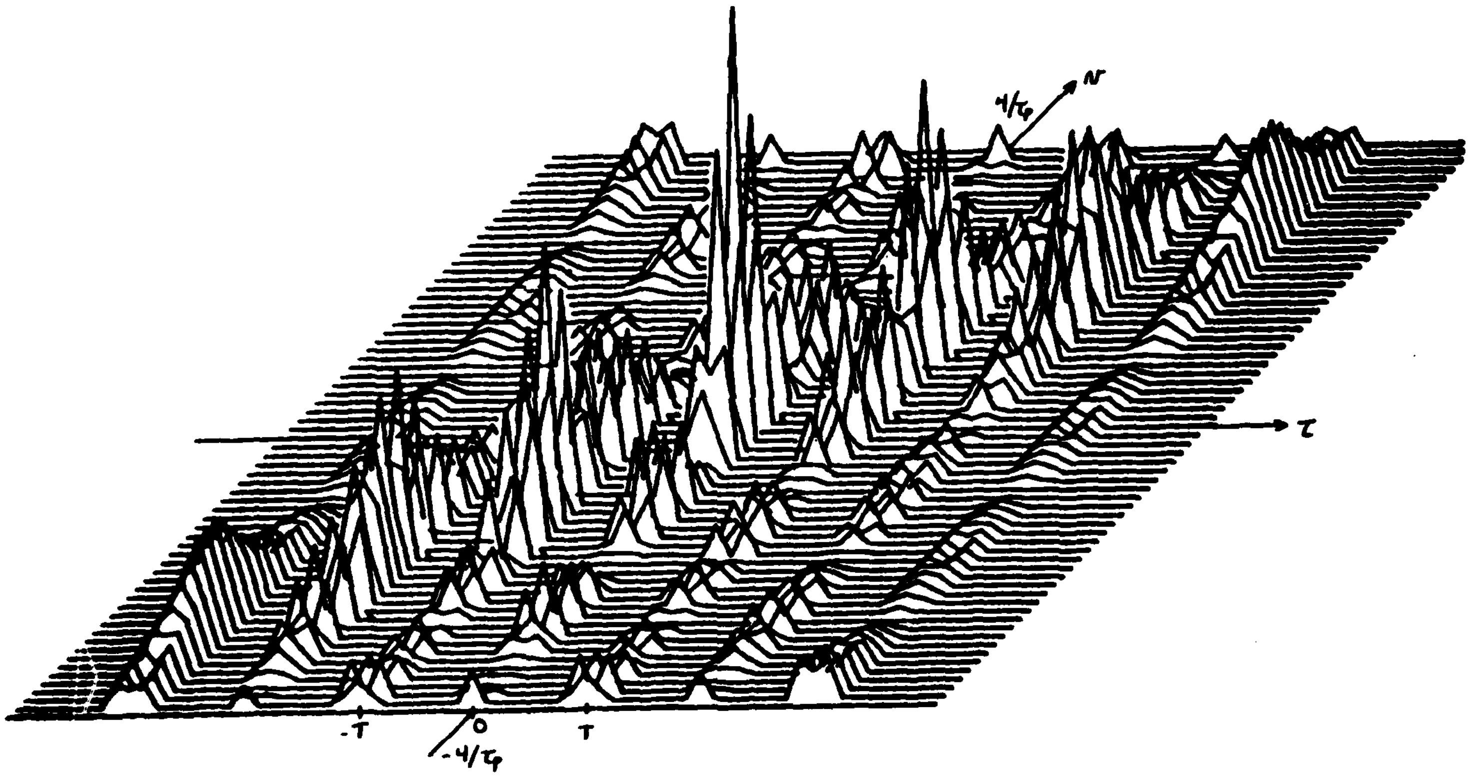


Figure 16. $|\chi(\tau,\nu)|$ for the Basic LSF Pulsetrain, N=4, T/t_p =4.

the doppler axis and it shows the peak amplitude of the p surfaces decreasing for increasing p. The p surfaces are seen to be shifted away from the delay axis as expected.

A plot of $\chi(\tau,\nu)$ for a two pulse waveform is shown in Figure 17. The characteristics of Figure 16 are again evident. Tables 8 and 9 show the symmetry expected of $\chi(\tau,0)$ and $\chi(0,\nu)$ respectively. Figure 18 plots the data of Table 8. Table 10 lists the magnitudes from column 16. It can be seen that the maximum of the p surface in the $\chi(-\tau,-\nu)$ quadrant is located at ν= -1000Hz (-6283 radians), and that the maximum value is 99; both the position and the value of this shifted p surface are as expected. The results then for the gross structure as as expected.

The range of the doppler shifts is next reduced to examine the central p surface. As in the constant carrier case, the doppler spread is set to +/- 1000 Hz, but with very different results. As seen in Figure 19, the p surfaces to either side of the doppler axis are greatly reduced in amplitude due the their shift away form the delay axis. A shearing or twisting of the peaks within the central ridge is also evident. The shearing is a result of the linear frequency shifting and is reminiscent of that for the linear FM waveform depicted in Figure 17. The magnitude data for $\chi(0,\nu)$ listed in Table 11 retains the symmetry expected. Figure 20 shows a five level contour wherein the twisting of the peaks in the central surface is evident. Its probable that these sheared ridges run parallel to the line given by $\tau/\nu = - T/f_r$ as would be the case for a single pulse linear FM waveform, but data was not collected to confirm this.

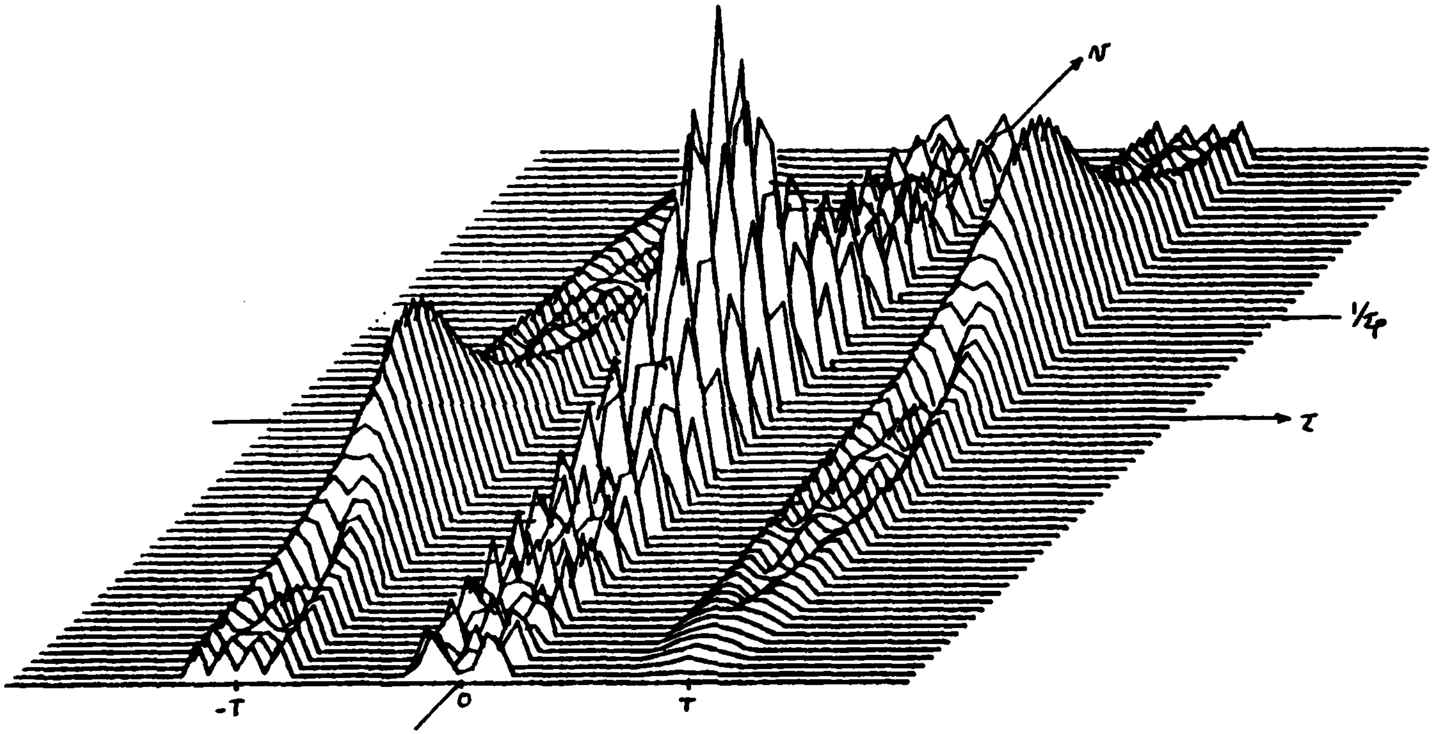


Figure 17. $|\chi(\tau,\nu)|$ for the Basic LSF Pulsetrain, N=2, T/t_p =4.

TABLE VIII. Magnitude Data for $\chi(\tau,0)$ of a LSF Pulsetrain, N=2.

****** DATA FOR ROW 32 *****

MAXIMUM VALUE= 32.000 AT TAU= .00000 (SAMPLE 128)
-3DB VALUE= 12.223 AT TAU= .00031 (SAMPLE 133)

1-32	33-64	65-96	97-128	129-160	161-192	193-224	225-256
.000	.000	1.000	.000	29.423	.000	1.000	.000
.000	.000	1.962	.000	25.869	.000	1.962	.000
.000	.000	2.848	.000	21.618	.000	2.848	.000
.000	.000	3.625	.000	16.971	.000	3.624	.000
.000	.000	4.262	.000	12.223	.000	4.262	.000
.000	.000	4.736	.000	7.654	.000	4.736	.000
.000	.000	5.027	.000	3.512	.000	5.027	.000
.000	.000	5.126	.000	.000	.000	5.126	.000
.000	.000	5.027	.000	2.731	.000	5.027	.000
.000	.000	4.736	.000	4.592	.000	4.736	.000
.000	.000	4.262	.000	5.556	.000	4.262	.000
.000	.000	3.624	.000	5.657	.000	3.624	.000
.000	.000	2.848	.000	4.989	.000	2.848	.000
.000	.000	1.962	.000	3.696	.000	1.962	.000
.000	.000	1.000	.000	1.962	.000	1.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	1.000	.000	1.961	.000	1.000	.000	.000
.000	1.962	.000	3.695	.000	1.962	.000	.000
.000	2.848	.000	4.989	.000	2.848	.000	.000
.000	3.624	.000	5.657	.000	3.625	.000	.000
.000	4.262	.000	5.556	.000	4.262	.000	.000
.000	4.736	.000	4.592	.000	4.736	.000	.000
.000	5.027	.000	2.731	.000	5.027	.000	.000
.000	5.126	.000	.000	.000	5.126	.000	.000
.000	5.027	.000	3.512	.000	5.027	.000	.000
.000	4.736	.000	7.654	.000	4.736	.000	.000
.000	4.262	.000	12.222	.000	4.262	.000	.000
.000	3.625	.000	16.970	.000	3.624	.000	.000
.000	2.848	.000	21.618	.000	2.848	.000	.000
.000	1.962	.000	25.869	.000	1.962	.000	.000
.000	1.000	.000	29.423	.000	1.000	.000	.000
.000	.000	.000	32.000	.000	.000	.000	.000

TABLE IX. Magnitude Data for $\chi(\tau,\nu)$ of a LSF Pulsetrain, N=2.

COLUMN #32 (TAU = .000000)

MAX VALUE = 199 AT W= .000 (SAMPLE #32)
-3dB VALUE= 75 AT W= 589.047 (SAMPLE #33)

DOPPLER SHIFT	MAGNITUDE	DOPPLER SHIFT	MAGNITUDE
-18260.500	2	589.047	75
-17671.460	9	1178.094	133
-17082.410	17	1767.145	161
-16493.360	0	2356.191	0
-15904.310	24	2945.242	125
-15315.260	18	3534.289	78
-14726.210	9	4123.340	32
-14137.170	20	4712.387	60
-13548.120	5	5301.434	13
-12959.070	4	5890.484	9
-12370.020	3	6479.531	5
-11780.970	0	7068.582	0
-11191.930	21	7657.629	30
-10602.870	22	8246.680	28
-10013.830	14	8835.727	16
-9424.777	43	9424.777	43
-8835.730	16	10013.820	14
-8246.684	28	10602.870	22
-7657.633	30	11191.920	21
-7068.586	0	11780.970	0
-6479.535	5	12370.020	3
-5890.488	9	12959.070	4
-5301.437	13	13548.120	5
-4712.391	60	14137.160	20
-4123.344	32	14726.210	9
-3534.293	78	15315.260	18
-2945.246	125	15904.310	24
-2356.195	0	16493.360	0
-1767.148	161	17082.410	17
-1178.098	133	17671.460	9
-589.051	75	18260.500	2
.000	199	18849.550	0

WAVEFORM PARAMETERS: # OF PULSES= 2 PULSEWIDTH= .001000
PRT= .004000 # OF SAMPLES= 16 SPACING= .000062500

DOPPLER PARAMETERS: 1ST FREQ= -18849.6
LAST FREQ= 18849.6 FREQ INCREMENTS= 589.04860

INITIAL PHASES: OPTION DECLINED (ALL SET TO ZERO)

CARRIER DRIFT: OPTION DECLINED (ALL SET TO ZERO)

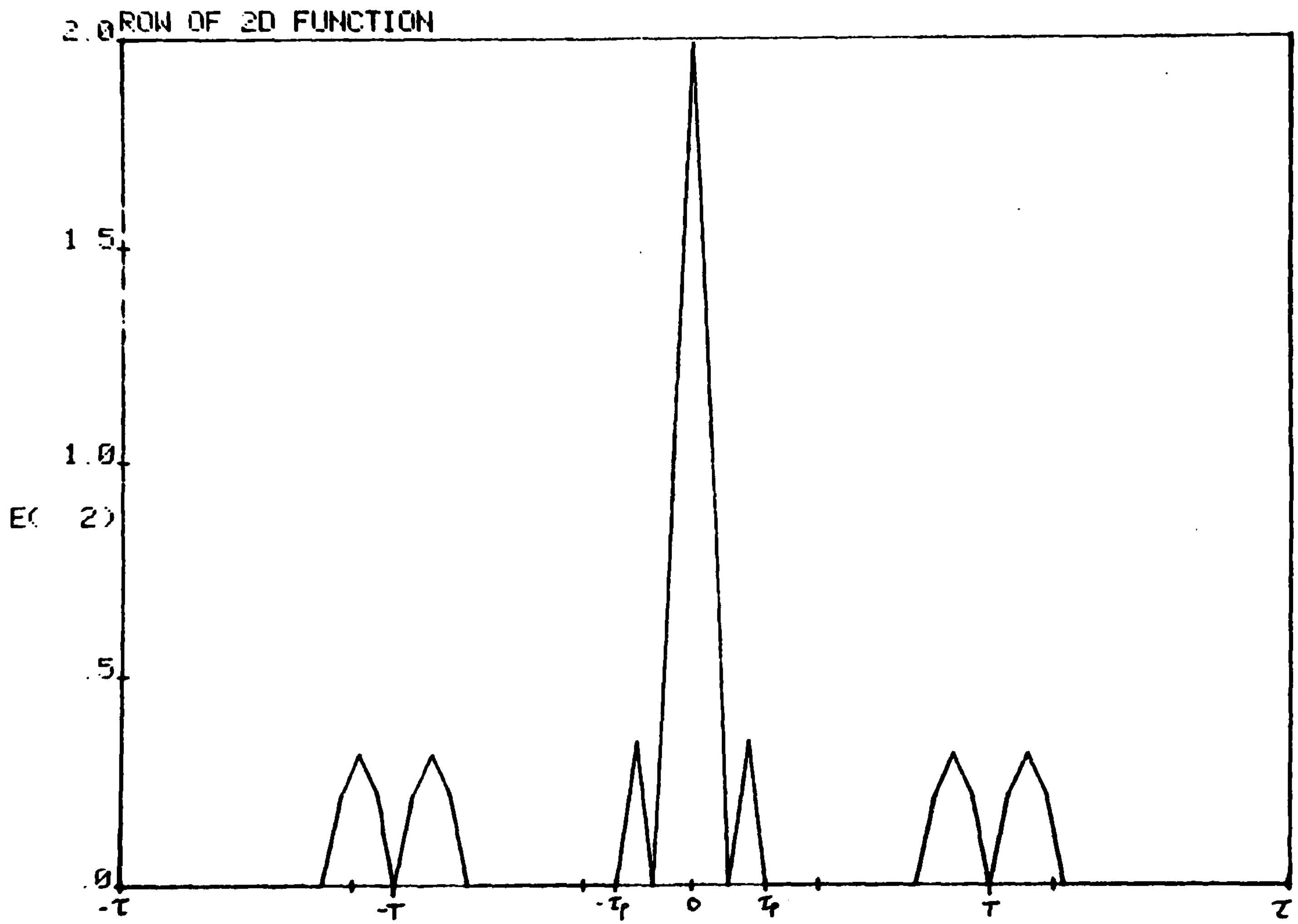


Figure 18. $|\chi(\tau,0)|$ for the Basic LSF Pulsetrain, N=2, T/t_p =4.

TABLE X. Magnitude Data for $\chi(\tau,\nu)$ of a LSF Pulsetrain, N=2, Column 16.

COLUMN #16 (TAU = -.008000)

MAX VALUE = 99 AT W= -6728.039 (SAMPLE #21)
-3dB VALUE= 46 AT W= -2346.992 (SAMPLE #49)

DOPPLER SHIFT	MAGNITUDE	DOPPLER SHIFT	MAGNITUDE
-9857.359	54	-4850.449	91
-9700.895	58	-4693.984	89
-9544.430	61	-4537.520	87
-9387.961	64	-4381.051	85
-9231.496	67	-4224.586	83
-9075.031	70	-4068.121	80
-8918.562	73	-3911.652	78
-8762.098	76	-3755.187	75
-8605.633	79	-3598.723	72
-8449.164	81	-3442.254	69
-8292.699	84	-3285.789	66
-8136.234	86	-3129.324	63
-7979.766	88	-2972.855	60
-7823.301	90	-2816.391	57
-7666.836	92	-2659.926	53
-7510.371	93	-2503.457	50
-7353.902	95	-2346.992	46
-7197.437	96	-2190.527	43
-7040.973	97	-2034.062	40
-6884.504	98	-1877.594	36
-6728.039	99	-1721.129	33
-6571.574	99	-1564.664	29
-6415.105	99	-1408.195	26
-6258.641	99	-1251.730	23
-6102.176	99	-1095.266	20
-5945.707	99	-938.797	17
-5789.246	98	-782.332	13
-5632.781	98	-625.867	10
-5476.312	97	-469.398	8
-5319.848	96	-312.934	5
-5163.383	94	-156.469	2
-5006.914	93	.000	0

WAVEFORM PARAMETERS: # OF PULSES= 2 PULSEWIDTH= .001000
PRT= .004000 # OF SAMPLES= 16 SPACING= .000062500

DOPPLER PARAMETERS: 1ST FREQ= -10013.8
LAST FREQ= .0 FREQ INCREMENTS= 156.46610

INITIAL PHASES: OPTION DECLINED (ALL SET TO ZERO)

CARRIER DRIFT: OPTION DECLINED (ALL SET TO ZERO)

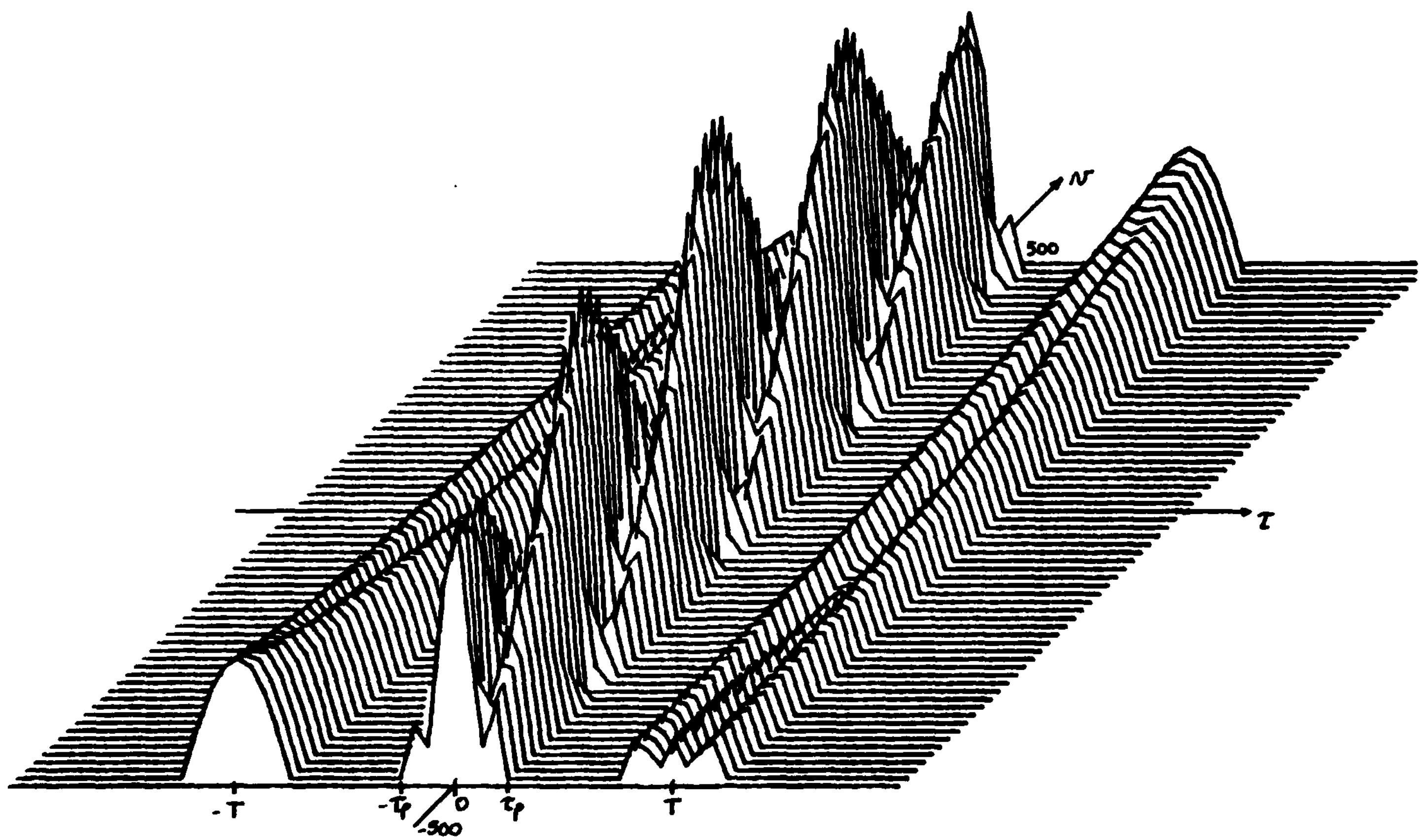


Figure 19. $|\chi(\tau,\nu)|$ for the Basic LSF Pulse Train,
$N=2$, $T/t_p = 4$, $|\nu|_{max} = 500$ Hz

TABLE XI. Magnitude Data for $\chi(0,\nu)$ of a LSF Pulse Train, N=2, $|\nu|_{max} = 1/t_p$.

COLUMN #32 (TAU = .000000)

MAX VALUE = 199 AT W= -.001 (SAMPLE #32)
-3dB VALUE= 75 AT W= 589.047 (SAMPLE #38)

DOPPLER SHIFT	MAGNITUDE	DOPPLER SHIFT	MAGNITUDE
-3043.418	128	98.174	196
-2945.243	125	196.348	184
-2847.069	115	294.523	165
-2748.894	101	392.698	140
-2650.719	81	490.873	110
-2552.544	57	589.047	75
-2454.370	29	687.222	38
-2356.195	0	785.397	0
-2258.020	31	883.572	37
-2159.845	62	981.743	73
-2061.671	92	1079.919	105
-1963.496	119	1178.095	133
-1865.321	143	1276.271	155
-1767.146	161	1374.442	170
-1668.972	174	1472.618	178
-1570.797	180	1570.794	180
-1472.622	178	1668.966	174
-1374.448	170	1767.142	161
-1276.273	155	1865.317	143
-1178.098	133	1963.493	119
-1079.923	105	2061.665	92
-981.749	73	2159.841	62
-883.574	37	2258.017	31
-785.399	0	2356.192	0
-687.224	38	2454.364	29
-589.050	75	2552.540	57
-490.875	110	2650.716	31
-392.700	140	2748.892	101
-294.525	165	2847.063	115
-196.351	184	2945.239	125
-98.176	196	3043.415	128
-.001	199	3141.591	127

WAVEFORM PARAMETERS: # OF PULSES= 2 PULSEWIDTH= .001000
PRT= .004000 # OF SAMPLES= 16 SPACING= .000062500

DOPPLER PARAMETERS: 1ST FREQ= -3141.6
LAST FREQ= 3141.6 FREQ INCREMENTS= 98.17474

INITIAL PHASES: OPTION DECLINED (ALL SET TO ZERO)

CARRIER DRIFT: OPTION DECLINED (ALL SET TO ZERO)

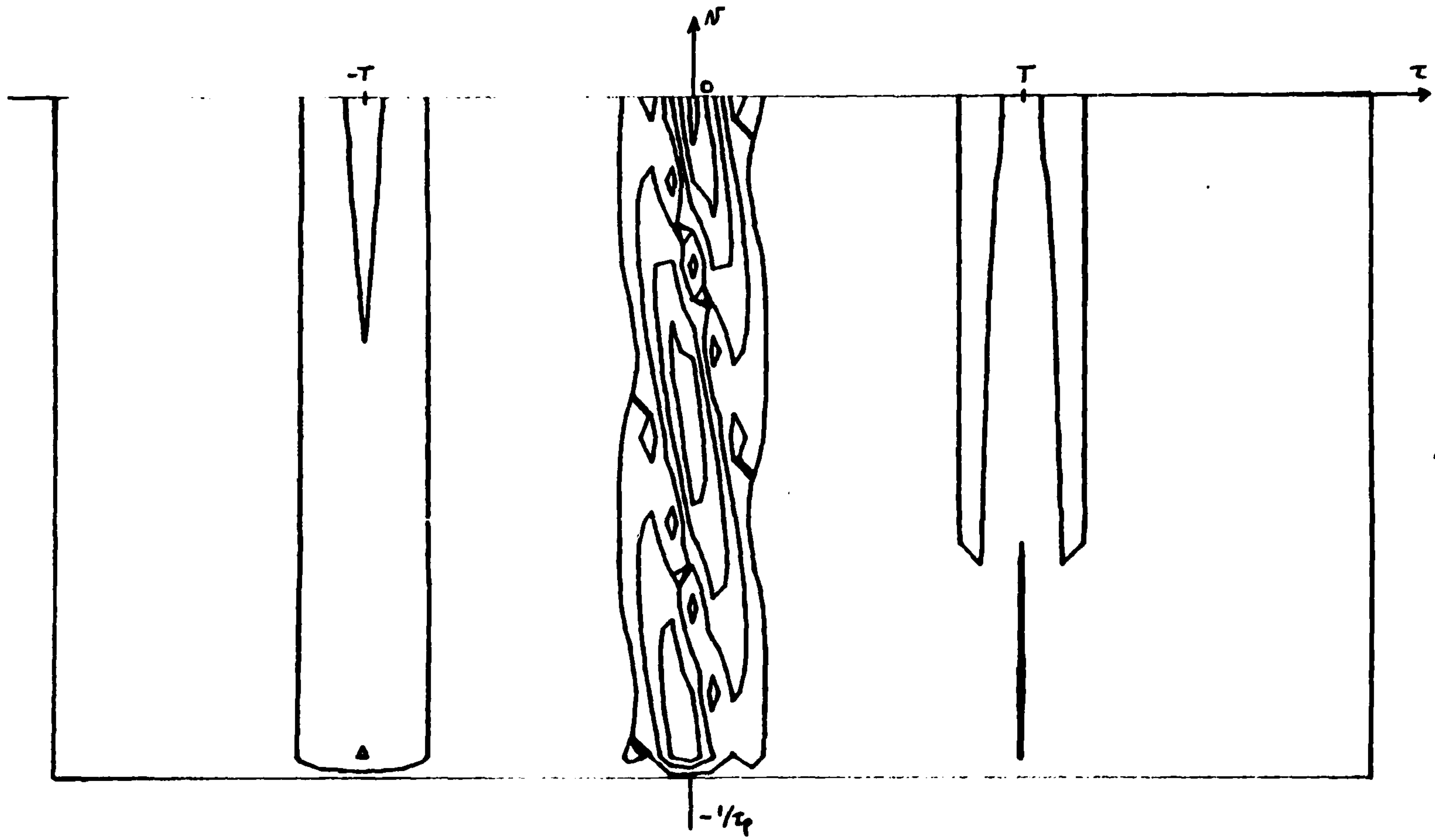


Figure 20. 5 Level Contour of $|\chi(\tau,\nu)|$,
$N=2$, $T/t_p=4$, $-500<\nu<0$.

Overall, the results obtained from the software are consistent with known or expected results. In the next chapter, the effects of random initial phases and random carrier drifts are examined.

V. Effects of Random Phases and Carrier Drifts

Resolution of the Basic Waveform

The first step in examining the effects of random initial phases and carrier drifts was to determine the range and doppler resolution of the undistorted waveform. The parameters of the pulsetrain used are as follows: pulsewidth, 0.001 second; pulse repetition time, 0.004 second; number of pulses, two; frequency step size, 1000 Hz. While not representative of a realistic waveform, these values were nonetheless used throughout what follows in order to minimize program execution time and keyboard input.

The delay resolution was calculated from the values listed in Table 12 for $\chi(\tau,0)$. A maximum value of 32.0 occurs at sample number 132. Since a value of 16.0 corresponding to the -6dB point is not explicitly listed, linear extrapolation was used on both sides of the maximum and the average taken as the delay resolution. Note that in order to calculate the resolution, the time interval between samples listed in Table 13 as 0.0000625 second must be utilized.

The delay resolution value, $\Delta\tau$,was calculated to be 0.0002627751 second. This may be expressed in terms of range as 39.42 kilometers (km) or 21.29 nautical miles (NM). Table 13 is a listing of $\chi(0,\nu)$ data for a narrow range of doppler shifts from which the doppler resolution was calculated. Again, linear extrapolation was used in calculations since a value of 16.0 is not listed. The doppler resolution, $\Delta\omega$, was calculated to be 520.32 radians/sec. Assuming a carrier frequency 100 times greater

TABLE XII. Delay Resolution Source Data for the Basic LSF Pulsetrain, N=2.

****** DATA FOR ROW 32-- W= .000 *****

MAXIMUM VALUE= 32.000 AT TAU= .00000 (SAMPLE 128)
-3DB VALUE= 12.223 AT TAU= .00031 (SAMPLE 133)

1-32	33-64	65-96	97-128	129-160	161-192	193-224	225-256
.000	.000	1.000	.000	29.423	.000	1.000	.000
.000	.000	1.962	.000	25.869	.000	1.962	.000
.000	.000	2.848	.000	21.618	.000	2.848	.000
.000	.000	3.625	.000	16.971	.000	3.624	.000
.000	.000	4.262	.000	12.223	.000	4.262	.000
.000	.000	4.736	.000	7.654	.000	4.736	.000
.000	.000	5.027	.000	3.512	.000	5.027	.000
.000	.000	5.126	.000	.000	.000	5.126	.000
.000	.000	5.027	.000	2.731	.000	5.027	.000
.000	.000	4.736	.000	4.592	.000	4.736	.000
.000	.000	4.262	.000	5.556	.000	4.262	.000
.000	.000	3.624	.000	5.657	.000	3.624	.000
.000	.000	2.848	.000	4.989	.000	2.848	.000
.000	.000	1.962	.000	3.696	.000	1.962	.000
.000	.000	1.000	.000	1.962	.000	1.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	1.000	.000	1.961	.000	1.000	.000	.000
.000	1.962	.000	3.695	.000	1.962	.000	.000
.000	2.848	.000	4.989	.000	2.848	.000	.000
.000	3.624	.000	5.657	.000	3.625	.000	.000
.000	4.262	.000	5.556	.000	4.262	.000	.000
.000	4.736	.000	4.592	.000	4.736	.000	.000
.000	5.027	.000	2.731	.000	5.027	.000	.000
.000	5.126	.000	.000	.000	5.126	.000	.000
.000	5.027	.000	3.512	.000	5.027	.000	.000
.000	4.736	.000	7.654	.000	4.736	.000	.000
.000	4.262	.000	12.222	.000	4.262	.000	.000
.000	3.625	.000	16.970	.000	3.624	.000	.000
.000	2.848	.000	21.618	.000	2.848	.000	.000
.000	1.962	.000	25.869	.000	1.962	.000	.000
.000	1.000	.000	29.423	.000	1.000	.000	.000
.000	.000	.000	32.000	.000	.000	.000	.000

TABLE XIII. Doppler Resolution Source Data for the LSF Pulsetrain, N=2.

COLUMN #32 (TAU = -.008000)

DOPPLER SHIFT	MAGNITUDE	DOPPLER SHIFT	MAGNITUDE
518.078	16.124	520.578	15.986
518.156	16.120	520.656	15.982
518.234	16.115	520.734	15.977
518.313	16.111	520.813	15.973
518.391	16.107	520.891	15.969
518.469	16.102	520.969	15.964
518.547	16.098	521.047	15.960
518.625	16.094	521.125	15.956
518.703	16.089	521.203	15.951
518.781	16.085	521.281	15.947
518.859	16.081	521.359	15.943
518.938	16.077	521.438	15.938
519.016	16.072	521.516	15.934
519.094	16.068	521.594	15.930
519.172	16.064	521.672	15.925
519.250	16.059	521.750	15.921
519.328	16.055	521.828	15.917
519.406	16.051	521.906	15.912
519.484	16.046	521.984	15.908
519.563	16.042	522.063	15.904
519.641	16.038	522.141	15.899
519.719	16.033	522.219	15.895
519.797	16.029	522.297	15.891
519.875	16.025	522.375	15.886
519.953	16.020	522.453	15.882
520.031	16.016	522.531	15.878
520.109	16.012	522.609	15.873
520.188	16.007	522.688	15.869
520.266	16.003	522.766	15.865
520.344	15.999	522.844	15.860
520.422	15.995	522.922	15.856
520.500	15.990	523.000	15.852

WAVEFORM PARAMETERS: # OF PULSES= 2 PULSEWIDTH= .001000
PRT= .004000 # OF SAMPLES= 16 SPACING= .000062500

DOPPLER PARAMETERS: 1ST FREQ= 518.0
LAST FREQ= 523.0 FREQ INCREMENTS= .07813

INITIAL PHASES: OPTION DECLINED (ALL SET TO ZERO)

CARRIER DRIFT: OPTION DECLINED (ALL SET TO ZERO)

than the frequency step size, $\Delta\omega$ corresponds to a radial velocity resolution of 1.242×10^5 meters per second (m/s), or 2.416×10^5 nautical miles per hour (knots/hr). In the sections to follow, the values calculated for $\Delta\tau$ and $\Delta\omega$ are used as standards for comparison.

Random Phase Effects

The effects of random phases were studied over two ranges: 0 to 20 degress, and 0 to 180 degrees. The first step in determining the values for $\Delta\tau$ and $\Delta\omega$ was to run CONPULS over a wide range of doppler shifts in order to estimate the position of the maximum value along the doppler axis. For the 20 degree range, the range was -64 to 64 radians/sec, and for the 180 degree range was -640 to 640 radians/sec. Then the program was run again over a much smaller range. Linear extrapolation was used to estimate the precise doppler value corresponding to the maximum value. However, the maximum value was found to occur over a doppler range of several radians if the total doppler spread was small. This made finding the exact center of the central lobe somewhat difficult. For purposes of this study, the value of doppler shift for which the data immediately to either side of sample number 128 ($\tau = 0$) exhibited symmetry was chosen as the center of the lobe. Once the doppler shift value corresponding to the center of the central peak was determined, subroutine TST was invoked to print all 256 magnitude values for this doppler value. The $\Delta\tau$ value is then calculated by extrapolation as in the basic waveform case. The program is then run over a narrow range of doppler shifts centered at approximately 520 radians greater than that

of the maximum value. Program GRAPH2 was then run to tabulate $\chi(0,\nu)$. The value of $\Delta\omega$ is then extrapolated from this data.

This process was repeated ten times for each phase range. The resultant effects on $\Delta\tau$ and $\Delta\omega$ for the 20 degree range are listed in Tables 14 and 15 respectively. The effects on $\Delta\tau$ and $\Delta\omega$ for the 180 range are presented in Tables 16 and 17 respectively.

Interpretation of Random Phase Results

The most noticeable effect of random phase errors is a translation of the entire ambiguity surface along the doppler axis. The greatest translation recorded was 510.5 radians in case number 9 for the 180 degree phase range, and is depicted in Figure 22. The shift is visible in comparison with Figure 19 which depicts the zero phase error case. In comparing the shift of the center value over both phase ranges, it is evident that greater shifts occur in the larger phase range. The summary of the effects for the 20 degree phase range presented in Table 15 shows that while the magnitude of the shifts varies considerably, the average shift tends towards zero. This observation is less evident in the 180 degree phase range results. Neither a qualitative nor a quantative relationship between the specific phases assigned to each pulse and the resultant shift along the doppler axis has been determined.

The effect on the delay and doppler resolution parameters, $\Delta\tau$ and $\Delta\omega$, are similar in the 20 and 180 degree phase range cases. In both, the delay resolution is equal to or greater than that for the zero degree phase case. Changes in $\Delta\tau$ are greater in the 180 degree phase range

TABLE XIV. Δt Results for 20 Degree Phase Range

Case #	Initial Phases $\mu(t)$ $\mu(-t)$	Δt Value x .001 sec	Δt Difference from Baseline Value (% - km - NM)
1	5.6,11.1 :9.7,10.1	.2627804	.00202 / .00605 / .00327
2	1.0,2.0 :9.7,10.1	.2627765	.00053 / .00160 / .00086
3	11.2,13.2:3.4,6.9	.2627751	.00000 / .00000 / .00000
4	2.7,5.3 :13.9,18.6	.2627751	.00000 / .00000 / .00000
5	8.9,8.6 :16.7,12.9	.2627751	.00000 / .00000 / .00000
6	7.1,5.0 :10.7,12.1	.2627751	.00000 / .00000 / .00000
7	14.7,8.9 :16.1,12.2	.2627844	.00354 / .01062 / .00573
8	0.8,1.6 :14.0,7.6	.2627817	.00251 / .00753 / .00407
9	16.4,12.4:15.6,10.7	.2627751	.00000 / .00000 / .00000
10	9.8,10.3 :12.2,13,2	.2627751	.00000 / .00000 / .00000

Summary : $\overline{\Delta t}$ = 0.0002627774 seconds

$\overline{\Delta t}$ difference: 0.00086% ; 0.00258 km ; 0.00139 NM

$\sigma^2(\Delta t)$: 0.0000015797% $\sigma(\Delta t)$: 0.0012569% ; 0.0037706 km; 0.0020365 NM

TABLE XV. $\Delta\omega$ Results for 20 Degree Phase Range

Case #	$\Delta\omega$ (rads/sec)	Shift of Center Value from ω=0	$\Delta\omega$ Difference From Baseline value (% - m/s - knots/hr
1	520.34	+22.0	.0038 / 4.77 / 9.29
2	520.29	+2.42	-.0058 / -7.16 / -13.93
3	520.35	-6.25	.0058 / 7.16 / 13.93
4	520.29	-8.59	-.0058 / -7.16 / -13.93
5	520.43	+14.73	.0211 / 26.26 / 51.08
6	520.71	-15.71	.0750 / 93.11 / 181.11
7	520.10	-53.01	.3415 / 424.22 / 825.19
8	520.40	+31.00	.0154 / 19.10 / 37.15
9	520.28	+3.87	-.0071 / 8.83 / 17.18
10	520.29	-1.88	-.0067 / -8.36 / -16.25

Summary: $\overline{\Delta\omega_o}$= -1.1426 rads/sec. $\overline{\Delta\omega}$= 520.34 rads/sec

$\overline{|\Delta\omega_o|}$= 15.92 rads.

$\overline{\Delta\omega}$ difference: 0.02 rads/sec ; 0.0038%; 4.77 m/s; 9.29 knots/hr

$\sigma^2(\Delta\omega)$: 8.35 rads/sec.

$\sigma(\Delta\omega)$: 2.89 rads/sec; 689.94 m/s ; 1342.04 knots/hr

TABLE XVI. Δt Results for 180 Degree Phase Range

Case #	Initial Phases $\mu(t)$: $\mu(-t)$	Δt Value x .001 sec	Δt Difference from Baseline Value (% - km - NM)
1	92.0,83.0 :103.0,123.0	.2628398	.02463 / .07398 / .03989
2	7.0,14.0 :144.0,104.0	.2629442	.06435 / .19305 / .10427
3	69.0,55.0 :120.0,157.0	.2629747	.07596 / .22789 / .12308
4	80.0,160.0 :91.0,79.0	.2633985	.23723 / .71171 / .38440
5	6.0,6.0 :160.0,132.0	.2628298	.02462 / .07387 / .03989
6	46.0,92.0 :154.0,124.0	.2632173	.16828 / .50484 / .27267
7	102.0,103.0:68.0,35.0	.2628534	.03166 / .09499 / .05130
8	105.0,127.0:156.0,128.0	.2629601	.07040 / .21120 / .11407
9	73.0,146.0 :57.0,13.0	.2638092	.39353 /1.18509 / .63764
10	83.0,65.0 :117.0,151.0	.2629761	.07649 / .22947 / .12394

Summary : $\overline{\Delta t}$ = 0.0002630803 seconds

$\overline{\Delta t}$ difference: 0.11671% ; 0.35014 km ; 0.18911 NM

$\sigma^2(\Delta t)$: 0.012620% $\sigma(\Delta t)$: 0.11234% ; 0.33701 km ; 0.18202 NM

TABLE XVII. $\Delta\omega$ Results for 180 Degree Phase Range

Case #	$\Delta\omega$ (rads/sec)	Shift of Center Value from ω=0	$\Delta\omega$ Difference From Baseline Value (% - m/s - knots/hr
1	521.89	-126.52	.3017 / 374.81 / 729.07
2	517.81	+205.00	-.4824 / -599.22 /-1165.58
3	523.11	-222.53	.5362 / 666.06 / 1295.60
4	515.36	+397.00	-.9533 /-1184.11 /-2303.29
5	518.78	+122.00	-.2960 / -367.65 / -715.13
6	516.15	+331.60	-.8014 / -995.50 /-1936.45
7	519.38	+148.37	-.1807 / -224.41 / -436.51
8	517.59	+218.16	-.5247 / -651.74 /-1267.74
9	513.84	+510.50	-1.2454 /-1546.99 /-3009.14
10	523.26	-226.99	.5650 / 701.87 / 1365.26

Summary: $\overline{\Delta\omega_o}$= 135.56 rads/sec. $\overline{\Delta\omega}$= 518.72 rads/sec.

$\overline{|\omega_o|}$= 250.6 rads/sec.

$\overline{\Delta\omega}$ difference: -1.60 rads/sec; 0.31 %; 318.97 m/s; 743.00 knots/hr

$\sigma^2(\Delta\omega)$: 9.42 rads/sec.

$\sigma(\Delta\omega)$: 3.07 rads/sec.; 732 m/s; 1425 knots/hr

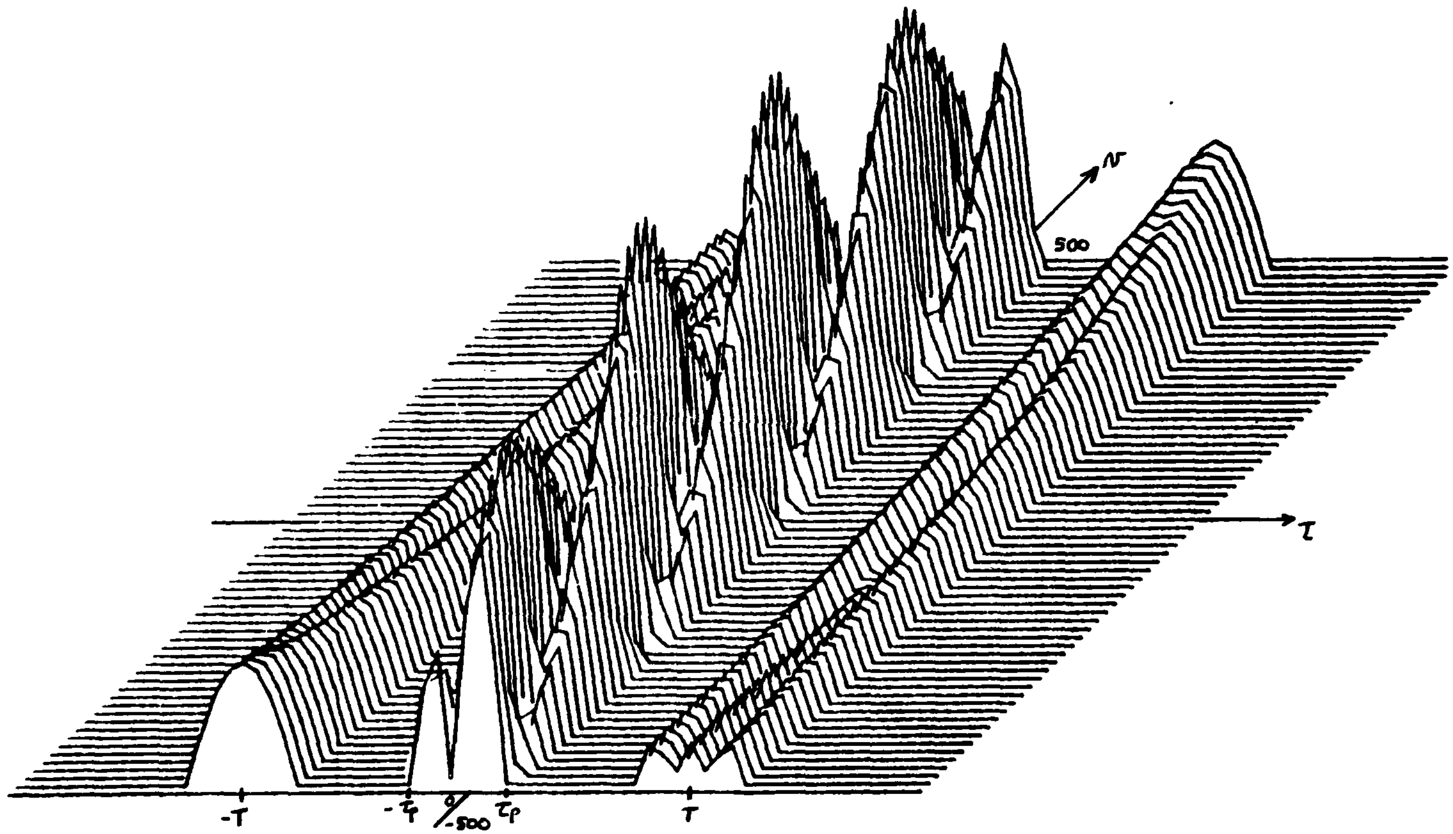


Figure 21. $|\chi(\tau,\nu)|$ for LSF Pulsetrain, with 180 Degree Phases, Case 9.

case. In the 20 degree phase range the delay resolution was uneffected in six of ten cases, and the greatest change was less than 0.0004 percent.

The effects on doppler resolution are similar to those on the delay resolution in that the greatest changes occur for the 180 degree phase range cases. However, the calculated doppler resolutions deviate about both sides of the zero degree phase case value as opposed to being only greater as was the case with delay resolution. That is both enhanced and degraded doppler resolution capabilities are recorded. The changes are small however, being no greater than 1.3 percent and generally considerably less. As was the case for the translation effect, a relationship between the specific random phases and the resulting resolution parameters is unknown.

The observed effect of both greater or reduced doppler resolution with a simultaneous decrease in delay resolution is interesting. At first, it would appear to violate the property that the volume under the ambiguity surface be constant. However, the property applies to the squared magnitude, $|\chi(\tau,\nu)|^2$, which does not take random phases into account. Since this study computes the magnitude only after the effects of random phases are introduced and summed, the constant volume property does not apply. Qualitatively, the effects of random phases on the resolution properties are negligible.

Effects of Random Carrier Drift

The effects of random carrier drift were examined for 0.1 and 1.0 percent carrier drift. For each percentage, the pulsewidth, pulse repetition time, the number of pulses, and the frequency step size was the same as in the random phase study. The carrier frequency was set to 100 times the frequency step. For each percentage, ten cases are examined. The $\Delta\tau$ and $\Delta\omega$ results for the 0.1% cases are listed in Tables 18 and 19. Results for 1.0% carrier drift are presented in Tables 20 and 21.

Interpretation of Carrier Drift Results

The effects of random carrier drift parallel those of random phases. A translation of the entire surface occurs along the doppler axis with the effect more pronounced for the larger carrier drift range. The greatest translation recorded was -4811 rads/sec in case 8, 180 degree range. The effect is presented graphically in Figures 22 and 23. Figure 22 depicts the response for case 8 over a range of plus and minus the frequency step size of 1000 Hz. Comparison of this response with that of the zero percent case (Figure 19), shows a large shift of the entire surface in the negative doppler direction. Figure 23 depicts the response centered at the doppler value corresponding to the maximum response. In the 0.1% drift range, comparison of the mean shift size with the mean of the shifts would indicate that the mean of the shifts tends towards zero. However, limited data coupled with less clear results for for the 1.0% range preclude establishing this trend with certainty.

TABLE XVIII. Δt Results for 0.1 % Carrier Drift

Case #	Carrier Drift (Hz) $\mu(t)$ $\mu(-t)$	Δt Value x .001 sec	Δt Difference from Baseline Value (% - km - NM)
1	22.2,44.4 : 86.5,73.0	.2624119	-.13821 / -.41465 / -.22395
2	46.6,42.6 : 25.0,50.0	.2608925	-.71643 /-2.14929 / -1.16084
3	54.4,57.9 : 75.6,51.2	.2648018	.77126 / 2.31380 / 1.24969
4	22.1,44.2 : 75.0,50.0	.2633315	.21117 / .63351 / .34216
5	24.6,49.2 : 73.6,47.2	.2631864	.15652 / .46956 / .25361
6	63.3,26.6 : 7.8,15.6	.2658257	1.16091 / 3.48275 / 1.88104
7	44.8,88.8 : 9.3,18.6	.2582107	-1.73699/-5.21100 / -2.81477
8	44.6,89.2 : 9.0,18.0	.2582252	-1.73148/-5.19444 / -2.80553
9	83.5,67.0 : 40.7,30.5	.2655632	1.06102 / 3.18306 / 1.71918
10	9.0,18.0 : 59.4,67.9	.2615297	-.47394 /-1.42182 / -.76793

Summary : $\overline{\Delta t}$ = 0.0002623979 seconds

$|\overline{\Delta t \text{ difference}}|$: 0.82 % ; 2.46 km ; 1.33 NM

$\sigma^2(\Delta t)$: 0.97035% $\sigma(\Delta t)$: 0.98506% 2.95518 km 1.59610 NM

TABLE XIX. $\Delta\omega$ Results for 0.1% Carrier Drift

Case #	$\Delta\omega$ (rads/sec)	Shift of Center Value from ω=0	$\Delta\omega$ Difference From Baseline Value (% - m/s - knots/hr
1	516.41	+26.25	-.7515 / -933.44 /-1815.70
2	521.10	-21.60	.1499 / 186.21 / 362.21
3	519.54	+20.46	-.1499 / -186.21 / -362.21
4	517.62	+34.60	-.5189 / -644.58 /-1253.81
5	516.16	+37.47	-.7995 / -993.13 /-1931.79
6	522.34	-32.50	.3882 / 482.42 / 938.03
7	523.94	+26.06	.6957 / 864.42 / 1681.03
8	524.00	+26.40	.7073 / 878.54 / 1708.90
9	524.32	-4.72	.7688 / 954.93 / 1857.49
10	516.66	0.00	-.7034 / -873.76 /-1699.61

Summary : $\overline{\Delta\omega_o}$= 11.24 rads/sec. $\overline{\Delta\omega}$= 520.18 rads/sec.

$|\overline{\Delta\omega}$ difference$|$: 2.93 rads/sec; 0.56 %; 699.72 m/s; 1361.07 knots/hr

$\sigma^2(\Delta\omega)$: 53.20 rads/sec.

$\sigma(\Delta\omega)$: 7.29 rads/sec.; 1740 m/s; 3385 knots/hr

TABLE XX. Δt Results for 1.0 % Carrier Drift

Case #	Carrier Drift (Hz) μ(t) : μ(-t)	Δt Value x .001 sec	Δt Difference from Baseline Value (% - km - NM)
1	183,366 : 785,570	.2727144	3.782 / 11.347 / 6.219
2	711,422 : 80,160	.2901621	10.422 / 31.267 / 16.887
3	208,416 : 781,562	.2725264	3.711 / 11.133 / 6.013
4	709,418 : 89,178	.2898676	10.310 / 30.930 / 16.706
5	435,379 : 267,534	.2537287	-3.443 /-10.328 / -5.578
6	692,384 : 161,322	.2890761	10.009 / 30.207 / 16.218
7	212,424 : 546,583	.2419481	-7.926 /-23.777 /-12.842
8	42,84 : 686,863	.2443974	-6.994 /-20.9814 /-11.332
9	327,145 : 549,589	.2794453	6.344 / 19.032 / 10.279
10	446,383 : 383,257	.2831507	7.754 / 23.262 / 12.564

Summary : $\overline{\Delta t}$ = 0.0002717017 seconds

$|\overline{\Delta t \text{ difference}}|$ = .0000185768 sec; 7.07%; 5.57 km; 3.01 NM

$\sigma^2(\Delta t)$: 45.174% $\sigma(\Delta t)$: 6.721 % ; 20.16 km ; 10.89 NM

TABLE XXI. Δω Results for 1.0% Carrier Drift

Case #	Δω (rads/sec)	Shift of Center Value from ω=0	Δω Difference From Baseline Value (% - m/s - knots/hr)
1	524.40	-2848.55	.7841 / 974.03 / 1894.64
2	521.10	2869.90	.1499 / 186.21 / 362.21
3	513.80	-2872.20	-1.2500 /-1566.54 /-3027.71
4	520.86	2861.77	.1038 / 128.92 / 250.76
5	514.53	1332.97	-1.1128 /-1382.26 /-2688.72
6	522.35	1225.47	.3901 / 484.62 / 942.68
7	519.20	-1441.95	-.2153 / -267.80 / -520.10
8	516.51	-4811.79	-.7324 / -909.57 /-1769.26
9	516.99	-1734.24	-.6400 / -794.98 /-1546.36
10	527.10	46.40	1.3030 / 1618.61 / 3148.45

Summary : $\overline{\Delta\omega}_o$= -532.7 rads/sec. $\overline{\Delta\omega}$= 519.68 rads/sec.

$|\overline{\Delta\omega}$ difference$|$: 3.48 rads/sec; 0.67 %; 830.79 m/s; 1616.02 knots/hr

$\sigma^2(\Delta\omega)$: 20.81 rads/sec.

$\sigma(\Delta\omega)$: 4.56 rads/sec.; 1088 m/s; 2118 knots/hr

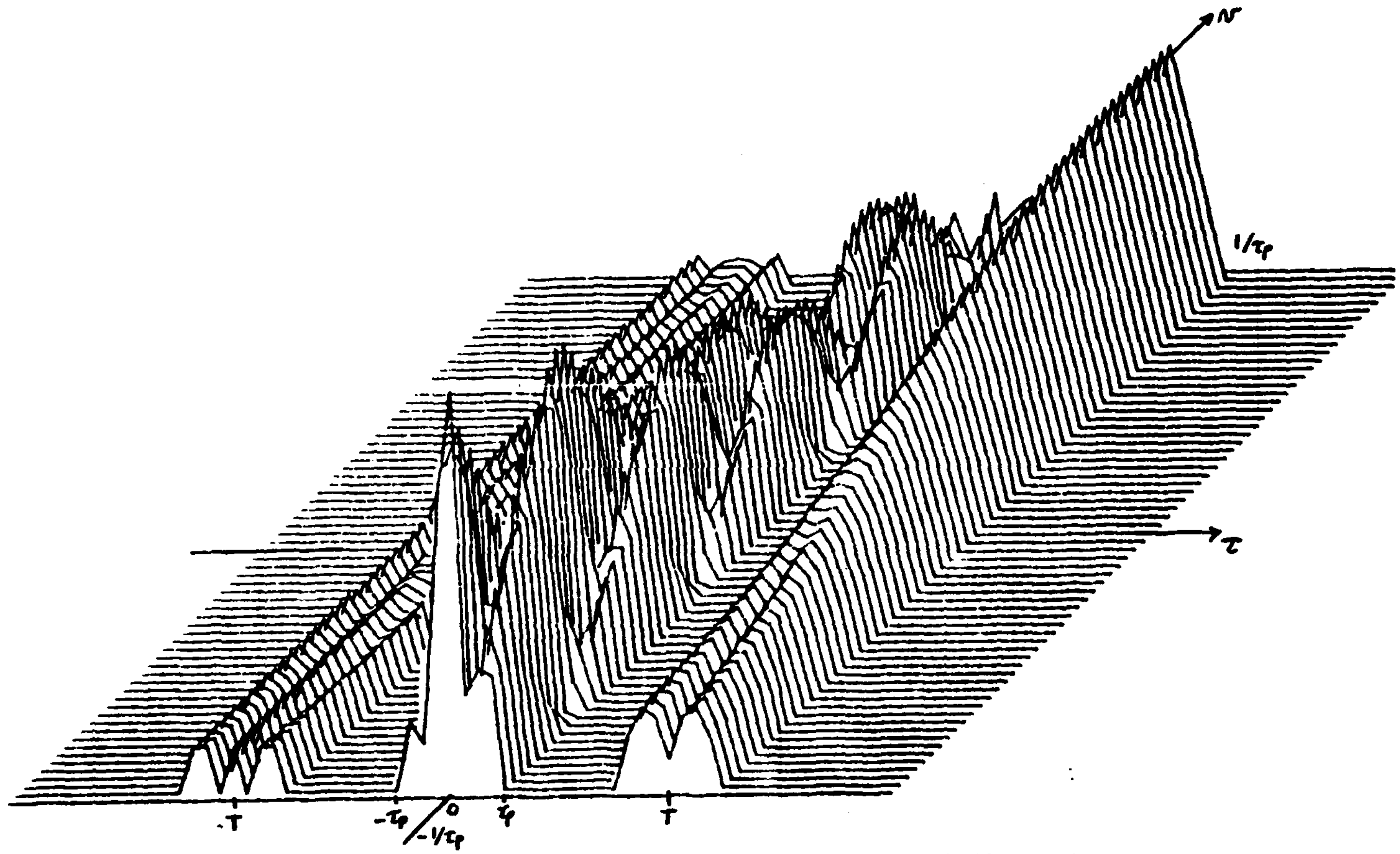


Figure 22. $|\chi(\tau,\nu)|$ for LSF Pulse Train, with 1.0% Carrier Drift, Case 8, $|\nu|_{max} = 500$ Hz

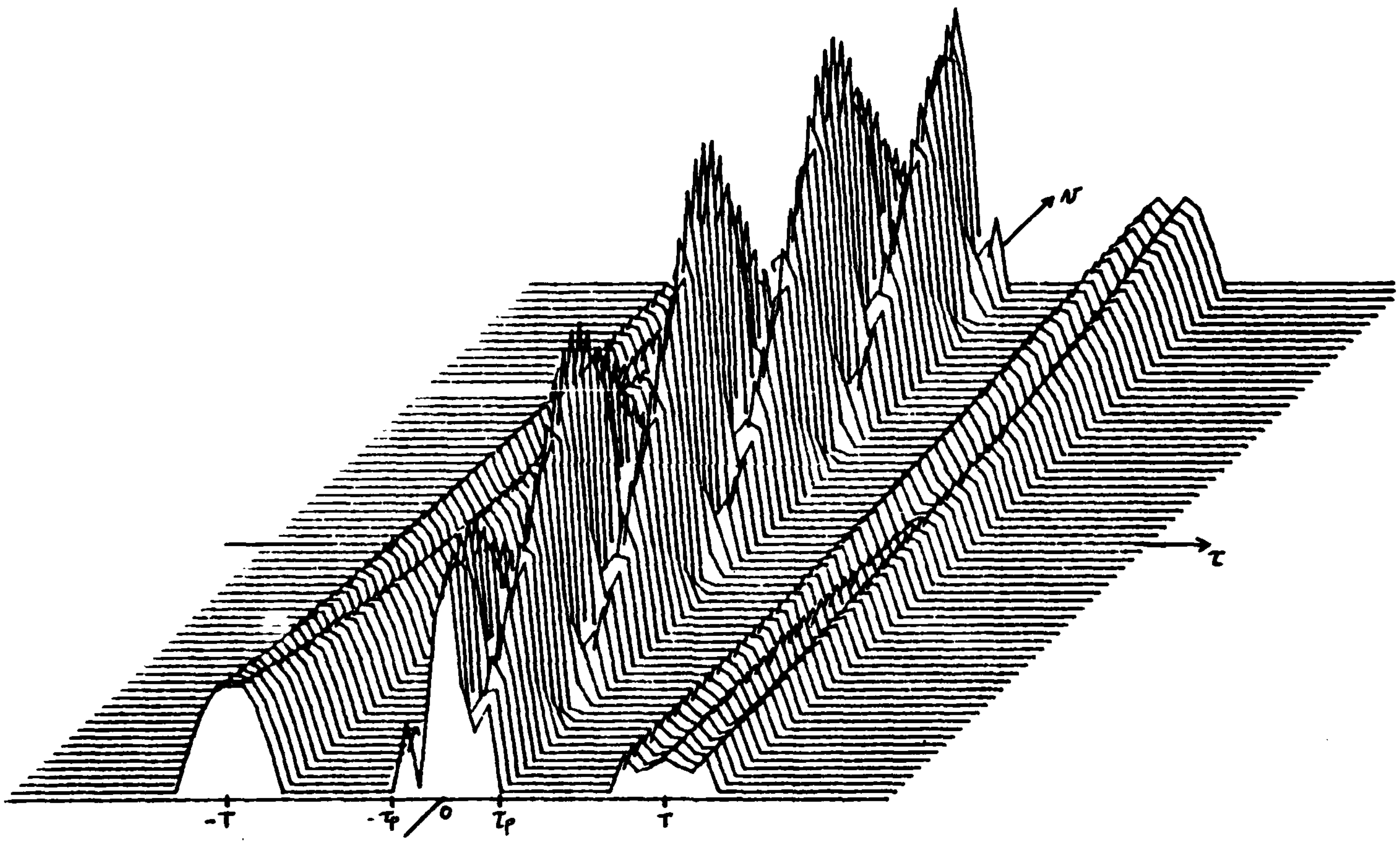


Figure 23. $|\chi(\tau,\nu)|$ for LSF Pulse Train, with 1.0% Carrier Drift, Case 8, ν = 765 +/- 500 Hz

In both drift ranges, the doppler resolution was observed to fluctuate around the zero percent drift case value, and greater deviations were observed for the larger drift range. The delay resolution results are unlike those in the random phase study. Recall that in the phase study, the delay resolution was always greater than or equal to the baseline resolution. Not so for random carrier drifts. The delay resolution is calculated to deviate about both sides of the baseline value as does the doppler resolution. As in the study of random phases, neither a qualitative nor quantitative relationship between specific carrier drifts the pulses and the translation, doppler and delay effects has been determined. Only the observation that the delay and doppler resolution changes are minimal can be made with confidence.

VI. Conclusions and Recommendations

Conclusions

The objective of this study has been to determine the effects of random initial phases and random carrier drifts on the range and doppler resolution properties of a linear step frequency pulse train. When the resolution parameters are defined as the half power point of the matched filter response, the effects on a two pulse train are shown to be minimal for phase ranges of 20 and 180 degrees and for carrier drifts of 0.1 and 1.0 percent. This conclusion is based on examination of data collected by computing the ambiguity function of the two pulse waveform.

The effects of random initial phases and carrier drifts on the resolution properties of longer pulse trains was not determined. However, the program developed provides the means to do so, as well as determining the effects of varying other pulse train parameters.

Recommendations

The translation of the ambiguity surface along the doppler axis is an interesting effect. A more detailed study could perhaps predict the value and direction of the shift for specific values of initial phase or carrier drift.

With modification, the software developed could compute and plot the ambiguity function for a pseudo random step frequency pulse train with staggered pulse repetition interval. Such a waveform is of perhaps

more interest than the basic linear step pulse train.

A final recommendation is that the software be adapted for the computers used within the Air Force Wright Aeronautical Laboratories Radar Group (AFWAL/AARM). There the software could prove to be an useful tool to the group's engineers in evaluation of future Air Force radar systems.

Bibliography

1. Beyer, William H. CRC Standard Mathematical Tables (27th edition). Boca Raton, FL: CRC Press, Inc., 1984

2. Bird, G.J.A. Radar Precision and Resolution. New York: John Wiley and Sons, 1974.

3. Cook, Charles E. and Bernfeld, Marvin. Radar Signals-an Introduction to Theory and Application. N.Y.: Academic Press, 1967.

4. Dorn, William S. and Greenberg, Herbert J. Mathematics and Computing: with Fortran Programming. New York: John Wiley and Sons, Inc., 1967.

5. Gaskill, Jack D. Linear Systems, Fourier Transforms and Optics. New York: John Wiley and Sons, 1978.

6. Mitchell, Richard L. Radar Signal Simulation. Dedham, MA: Artech House, Inc., 1976.

7. Nathanson, Fred E. Radar Design Principles. New York: McGraw-Hill Book Company, 1969.

8. Rader, C. M. IEEE Programs for Digital Signal Processing. New York: The IEEE Press, 1979.

9. Reed, Capt John D. Radar Waveform Selection Based on the Calculation and Application of Radar Ambiguity Functions. MS Thesis, GE/EE/82J-11. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, June 1982.

10. Rihaczek, August W. "Radar Waveform Selection-A Simplified Approach," IEEE Transaction on Aerospace and Electronic Systems, AES-7(6): 1078-1086 (November 1971).

11. Rihaczek, August W. Principles of High-Resolution Radar. New York: McGraw-Hill Book Company.

12. Rihaczek, August W. "Radar Resolution Properties of Pulse Trains," Proceedings of the IEEE, 52: 153-165 (Febrary 1964).

13. Skolnik, Merrill I. Introduction to Radar Systems (Second Edition). New York: McGraw-Hill Book Company, 1980.

14. Skolnik, Merrill I. Radar Handbook. New York: McGraw-Hill Book Company, 1970.

15. Woodard, P.M. Probaility and Information Theory, with Applica-ations to Radar (Second Edition). Oxford, England: Pergomon Press Ltd., 1964.

Appendix A

<u>Program Listings for CONPULS</u>

Appendix A contains the listings for program CONPULS and its subroutines: MSG1, INDAT, FERRORS, PERRORS, INITIAL, LDARYS, TST1, FOUREA, and SAVDAT. The subroutines were written as overlays to conserve RAM space during program execution. The corresponding overlay names are: INTRO, IN, FER, PER, INIT, LARS, TST, FOUR, and SAVE.

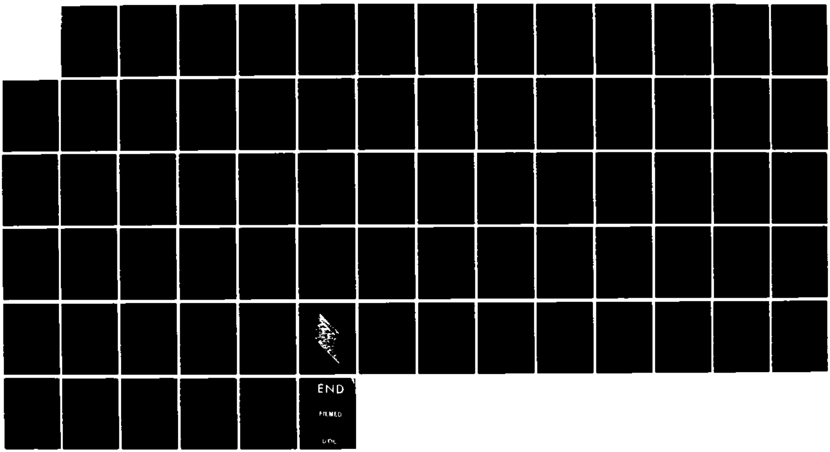
The source code was written, compiled, and linked on the Data General Eclipse S/250 computer located in the AFIT Signal Processing Laboratory. The operating system was General Data's RDOS, version 7.41. The source code was compiled by Data General's Fortran 5 compiler, RDLR, version 6.16 and linked by linker version 7.40.

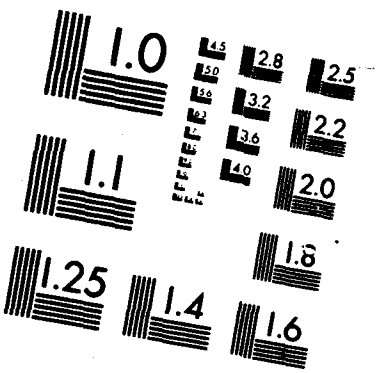
The source, object, and executable code was located on disk DP4 under directory EGRIFFIN. The entire directory has been written to magnetic tape for archival under the care of Mr. Dan Zambon, technician for the Eclipse computer system.

```
C***********************************************************************
C*                                                                     *
C*   PROGRAM CONPULS (CONVOLVE PULSES)                                 *
C*                                                                     *
C*   WRITTEN BY: CAPT THOMAS L. GRIFFIN JR                             *
C*   DATE:  10 AUGUST 1985                                             *
C*                                                                     *
C*   THIS PROGRAM COMPUTES THE AMBIGIUITY FUNCTION FOR A               *
C*   LINEAR STEP FREQUENCY PULSETRAIN.  THE PULSE ENVELOPE IS          *
C*   RECTANGULAR, THE REPETITON TIME IS UNIFORM, THE REPETITION RATE   *
C*   IS 2 OR 4 TIMES THE PULSEWIDTH, AND 2,4 OR 8 PULSES MAY BE        *
C*   SELECTED.  A NUMBER OF OPTIONS ARE AVAILABLE:  SIMULATE A         *
C*   CONSTANT CARRIER PULSE TRAIN, INCLUDE RANDOM INITIAL PHASES IN    *
C*   IN 4 RANGES, INCLUDE RANDOM CARRIER FREQUENCY DRIFTS OF UP TO     *
C*   1.0 PERCENT OF THE CARRIER FREQUENCY.  SEVERAL SUBROUTINES        *
C*   REQUIRE KEYBOARD INPUT OF VARIABLES.  INSTRUCTIONS ARE GIVEN AS   *
C*   NEEDED.  MOST VARIABLES ARE LISTED BELOW AND EXPLAINED IN MORE    *
C*   DETAIL CHAPTER 3 OR WITHIN COMMENT LINES.  REFER TO CHAPTERS      *
C*   2 AND 3 FOR ADDITIONAL DETAILS, INCLUDING THE EXPRESSION WHICH    *
C*   IS EVALUATED BY THIS PROGRAM.                                     *
C*                                                                     *
C*      PW = WIDTH OF EACH PULSE IN THE WAVEFORM.  RANGE VARIES        *
C*           FROM X.0 TO 0.00000X SECONDS.                             *
C*                                                                     *
C*      RT = PULSE PEPETITION TIME.  LIMITED TO 2 OR 4 TIMES THE       *
C*           PULSEWIDTH.                                               *
C*      NP = NUMBER OF PULSES IN THE WAVEFROM: 2,4,OR 8.               *
C*      WS = STARTING VALUE OF DOPPLER FREQUENCY SHIFT IN CALCULATING  *
C*           THE AMBIGUITY FUNCTION: IN RADIANS.                       *
C*      WF = FINAL VALUE OF DOPPLER FREQUENCY SHIFT EVALUATED.         *
C*                                                                     *
C*     THE VALUES ASSIGNED TO THESE VARIABLES ARE USED TO DETERMINE THE*
C*     FOLLOWING:                                                      *
C*                                                                     *
C*   RATIO = THE RATIO OF PRT TO PW.  USED AS A MULTIPLIER IN DETER-   *
C*           MINING THE POSITION OF THE TIME-DELAYED WAVEFORM.         *
C*    TIME = THE TIME INTERVAL BETWEEN SAMPLES OF EACH PULSE: USED IN  *
C*           "CONPULS" TO CALCUATE VALUES FOR TIME-VARYING PHASE TERMS.*
C*     NUM = NUMBER OF SAMPLES USED TO REPRESENT EACH PULSE: A POWER OF*
C*           TWO FROM 4 TO 64.                                         *
C*     WIN = FREQUENCY INCREMENT BETWEEN EACH DOPPLER SHIFT EVALUATED. *
C*    WCEN = CENTER OF THE DOPPLER FREQUENCY SHIFT RANGE.              *
C*   W(64) = AN ARRAY CONTAINING THE VALUE OF DOPPLER SHIFT WHICH WILL *
C*           BE USED FOR EACH ROW (CUT) OF THE AMBIGUITY FUNTION.      *
```

AD-A163 953 EFFECTS OF RANDOM PHASE SHIFTS AND CARRIER DRIFT ON THE RESOLUTION PROPER..(U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI.. T L GRIFFIN 2/2

UNCLASSIFIED DEC 85 AFIT/GE/ENG/85D-19 F/G 9/4 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
C*     SIZE = ASSIGNED VALUE OF 256: USED IN COMPUTING NUM.  256 IS THE *
C*            SIZE OF THE ARRAY IN WHICH EACH PULSE IS FOUND.           *
C*    LIMIT = A FUNCTION OF NP AND RATIO, USED IN COMPUTING NUM.        *
C*   CENTER = ASSIGNED VALUE OF 128, USED TO SET THE LEFT AND RIGHT     *
C*            EDGES OF EACH PULSE.                                      *
C*    START = ASSIGNED VALUE OF ZERO: PREPRESENTS THE EXTREME LEFT SIDE *
C*            OF THE ARRAY IN WHICH EACH PULSE IS FOUND.                *
C*     STOP = ASSIGNED VALUE OF 256: REPRESENTS THE EXTREME RIGHT SIDE. *
C*                                                                      *
C*    RATIONALE FOR THE RANGE AND LIMITS OF THESE VARIABLES IS DETAILED *
C*    IN CHAPTER II.  SOME RATIONALE IS GIVEN AS COMMENTS WITHIN THE    *
C*    ROUTINE.                                                          *
C*                                                                      *
C*    THE RDOS COMMAND TO COMPILE THE PROGRAM IS: FORTRAN CONPULS       *
C*                                                                      *
C*    THE RDOS COMMAND TO LINK THE PROGRAM IS:                          *
C*                                                                      *
C*    RLDR CONPULS [INTRO,IN,FER,PER,INIT,LARS,TST,FOUR,SAVE] @FLIB@    *
C*                                                                      *
C***********************************************************************

        EXTERNAL INTRO
        EXTERNAL IN
        EXTERNAL FER
        EXTERNAL PER
        EXTERNAL INIT
        EXTERNAL LARS
        EXTERNAL TST
        EXTERNAL FOUR
        EXTERNAL SAVE

        INTEGER LEFT,RIGHT,CENTER,RATIO,NUM,RANGE,
     *          NP,SIZE,LIMIT,START,STOP,TEST,TEST1,TEST2,TEST3,
     *          TESTP,TESTF,ROW
        COMPLEX X(256),Y(256),AMT1,AMT2,FACT1,FACT2,
     *          FACT3,AMT3,FACT4,FACT5,FACT6,Z(256)
        REAL    PW,PRT,PI,WI,FR,TIME,PHASE1,PHASE2,W(64),FACT,
     *          SCALE,PHASEA,PHASEB,PHASEC,FE1(8),FE2(8),PE1(8),
     *          PE2(8),FO,PHASE3,PHASED,PHASEE,PHASEF,RANS(64,64),
     *          IANS(64,64),ANS(128),TEMP(128),PERCNT


C***    THIS NEXT LINE OPENS THE OVERLAY CAPABILITY
```

```
      CALL OVOPN(4,"CONPULS.OL",IER)

C***  THE NEXT SECTION CALL THE OPENING MSG.  THIS IS WRITTEN THIS
C***  WAY TO CONSERVE MEMORY IN THE ECLIPSE

      CALL OVLOD(4,INTRO,1,IER)
      CALL MSG1

C***  THE SUBROUTINE INDAT (INPUT DATA) IS CALLED IN ORDER TO SPECIFY
C***  THE VARIABLES WITHIN THE PROGRAM

      CALL OVLOD(4,IN,1,IER)
      CALL INDAT(PW,PRT,RATIO,NP,SIZE,LIMIT,NUM,CENTER,START,
     *     STOP,WS,WF,WIN,WCEN,W,TIME,ROW,TEST2)

C***  THE FOLLOWING TWO CALLS TO SUBROUTINES SET UP THE CARRIER FREQ
C***  SHIFTS AND INITIAL PHASES TO BE USED.  SEE THE SEPARATE LISTING
C***  OF EACH FOR DETAILS.  THE SUBROUTINES ARE ACTUALLY COMPILED AND
C***  LINKED AS OVERLAYS TO CONSERVE SPACE IN THE ECLIPSE COMPUTER.
C***  THAT'S WHY THE CALL OVERLAY LOAD STATEMENTS.

      CALL OVLOD(4,FER,1,IER)
      CALL FERRORS(FO,FE1,FE2,NP,PW,TESTF,PERCNT)

      CALL OVLOD(4,PER,1,IER)
      CALL PERRORS(PE1,PE2,NP,TESTP,RANGE)

C***  THE NEXT SECTION CALLS SUBROUTINE "INITIAL".  THE ROUTINE IN-
C***  INTIALIZES SEVERAL VARIABLES: SCALE, PI, FR, AND ARRAYS RANS
C***  AND IANS.  PLUS THERE ARE SEVERAL MESSAGES THAT ARE DISPLAYED.
C***  THE ROUTINE WAS WRITTEN AS AN OVERLAY TO SAVE MEMORY SPACE.
C***  SEE A PRINTOUT OF "IN.FR" FOR DETAILS.

      CALL OVLOD(4,INIT,1,IER)
      CALL INITIAL(NUM,NP,SCALE,PI,TEST1,FR,RANS,IANS,PW)

C***  THE NEXT SECTION OF CODE BEGINS THE HEART OF THE PROGRAM.  IT
C***  IS A THREE-LEVEL DO-LOOP, IN WHICH ALL THE PULSES ARE CONVOLVED
C***  WITH ONE ANOTHER, AND SUMMED IN ONE ARRAY.  THIS IS ACCOMPLISHED
C***  FOR 64 VALUES OF DOPPLER SHIFT.  LINE 30 IS THIS BLOCK'S END.
C***  DO-LOOPS 30 AND 40 START THE COMPUTATIONS; THE LOOP CONTROL
C***  PARAMETERS "M" AND "N" FORM THE DELAY APPIED TO THE TIME
C***  REVERSED AND DELAYED WAVEFORM BEING USED IN THE CALCULATIONS;
C***  DO-LOOPS 30 AND 40 SEQUENCE THE LINEAR CONVOLUTION OF EACH PULSE
```

```
C***    IN THE PULSE TRAIN WITH THE EACH OTHER PULSE.  DO LOOP 50
C***    SEQUENCES THE CONVOLUTIONS THROUGH 64 VALUES OF DOPPLER SHIFT.
C***    TOTAL LENGTH OF DO-LOOP 30 IS 2*NP*64.  IN EACH LOOP, THREE
C***    256-POINT FFT'S ARE COMPUTED: TWO FORWARD AND ONE INVERSE.

        DO 30 M=0,NP-1
           WRITE(10,300)M
           DO 40 N=0,NP-1
              WRITE(10,310)N
              WRITE(10,320)M-N
              TYPE ' '

              DO 50 P=1,64
                 WRITE(10,330)W(P)

C***    SUBROUTINE LDARYS (LOAD ARRAYS X AND Y) IS CALLED FOR EACH VALUE
C***    OF DOPPLER SHIFT.  THE ROUTINE CREATES TWO FINITE SEQUENCES
C***    WHICH REPRESENT THE TWO PULSES TO BE CORRELATED (CONVOLVED IN THE
C***    FREQUENCY DOMAIN).  SEE SEPARATE LISTING FOR ADDITIONAL DETAILS.

                 CALL OVLOD(4,LARS,1,IER)
                 CALL LDARYS(W,CENTER,NUM,Y,X,PW,PI,M,N,FR,FE1,FE2,FO,
     *             RATIO,P,TIME)

C***   THE SEQUENCES REPRESENTING THE TWO PULSES TO BE CONVOLVED ARE NOW
C***   IN ARRAYS X AND Y, AND ARE READY FOR THEIR DISCRETE FOURIER
C***   TRANSFORMS TO BE TAKEN.   SUBROUTINE FOUREA IS CALLED.  SEE THE
C***   LISTING OF IT FOR DETAILS.  AFTER EACH TRANSFORM IS TAKEN, THE
C***   RESULTANT ARRAYS ARE MULTIPIED POINT-BY-POINT AS THE NEXT STEP IN
C***   THE CONVOLUTION, WITH RESULTS STORED IN THE X ARRAY.  THE INVERSE
C***   TRANSFORM OF THE X ARRAY IS THEN TAKEN TO RETURN TO THE TIME
C***   DOMAIN.  THEN THE RESULT IS ADDED POINT-BY-POINT TO PREVIOUS
C***   RESULTS.  FINALLY, DO-LOOP 50 IS INCREMENTED.

                 CALL OVLOD(4,FOUR,1,IER)
                 CALL FOUREA(X,SIZE,-1)

                 CALL OVLOD(4,FOUR,1,IER)
                 CALL FOUREA(Y,SIZE,-1)

C***   DO-LOOP 60 MULTIPLES POINT-BY POINT THE FREQUENCY COMPONENTS
C***   OF EACH PULSE.  THE MULTIPICATION IN THE FREQUENCY DOMAIN IS
C***   EQUIVALENT TO CONVOLUTION IN THE TIME DOMAIN.  THE LAST STEP
C***   IS TO COMPUTE THE INVERSE TRANSFORM OF THE PRODUCT.
```

```
                DO 60 I=1,256
                   X(I)=X(I)*Y(I)
60              CONTINUE

                CALL OVLOD(4,FOUR,1,IER)
                CALL FOUREA(X,SIZE,1)


C***    THE FOLLOWING LINES (THROUGH #80) ACCOUNT FOR THE WEIGHTING
C***    GIVEN TO EACH SUM OVER M AND N.  THE PHASE EXPRESSIONS ARE:
C***
C***    FACTOR1= EXP(j2*PI*M*FR*(M-N)T) WHERE T=THE PRT= RATIO*PW
C***    FACTOR2= EXP(jW*N*T)
C***    FACTOR3= EXP(j2*PI*FR*M*t) WHERE t=THE TIME AT WHICH EACH
C***                               SAMPLE OCCURS.
C***    FACTOR4=EXP(j2*PI*FE1*(FE2-FE1)*FO*M*RATIO*PW
C***    FACTOR5=EXP(j(PE2-PE1)
C***    FACTOR6=EXP(j2*PI*FE1*FO*TIME*I)
C***
C***    REFERENCE CHAPTER II FOR THE DERIVATION OF THESE EXPRESSIONS.

                PHASEA=2*PI*M*FR*((M-N)*RATIO*PW)
                PHASEB=W(P)*(N+1)*RATIO*PW
                PHASED=2*PI*FE1(M)*(FE2(N)-FE1(M))*FO*M*RATIO*PW
                PHASEE=PE2(N+1)-PE1(M+1)
                FACT1=CMPLX(0.0,PHASEA)
                FACT2=CMPLX(0.0,PHASEB)
                FACT4=CMPLX(0.0,PHASED)
                FACT5=CMPLX(0.0,PHASEE)

C***    DO LOOP 80 DISTRIBUTES THE WEIGHTING OF EACH POINT IN THE X
C***    ARRAY FOR ALL VALUES OF TAU.  THE FOUR FACTORS ABOVE ARE IN-
C***    DEPENT OF TAU, WHILE THE TWO IN THE LOOP ARE NOT.

                DO 80 I=1,256
                   PHASEC=2*PI*FR*(I-128)*TIME*M
                   PHASEF=2*PI*FE1(M)*FO*TIME*(I-128)
                   FACT3=CMPLX(0.0,PHASEC)
                   FACT6=CMPLX(0.0,PHASEF)
                   X(I)=X(I)*CEXP(FACT1+FACT2+FACT3+FACT4+FACT5+FACT6)
80              CONTINUE
```

```
C***    THE NEXT TEST DETERMINES IF THE PRINTING OF A ROW SHALL BE
C***    ACCOMPLISHED.  IT CALLS SUBROUTINE "TST" ONLY IF TEST2 EQUALS 1
C***    (INPUT FROM KEYBOARD IN SUBROUTINE "INPUT") AND P (DOPPLER
C***    SHIFT) EQUALS THE ROW SELECTED.  SEE PRINTOUT OF SUBROUTINE "TST"
C***    FOR DETAILS.

                  IF(TEST2.NE.1.OR.P.NE.ROW) GO TO 70
                  CALL OVLOD(4,TST,1,IER)
                  CALL TST1(X,M,N,NP,ROW,TIME,Z)

C***   THIS NEXT BLOCK WILL STORE THE RESULTS FOR THIS VALUE OF P.
C***   HOWEVER, SINCE THE PLOTTING PROGRAM TO BE USED WILL ONLY ACCEPT
C***   AN ARRAY 64X64, ONLY EVERY EIGTH POINT IS STORED.

70                DO 90 J=1,64
                     RANS(P,J)=RANS(P,J)+REAL(X(J*4))*SCALE
                     IANS(P,J)=IANS(P,J)+AIMAG(X(J*4))*SCALE
90                CONTINUE

50             CONTINUE    ; TO NEXT VALUE OF W (DOPPLER SHIFT)
40          CONTINUE       ; TO NEXT VALUE OF N
30       CONTINUE          ; TO NEXT VALUE OF M

C***    THIS PART COMPUTES THE MAGNITUDE FOR EACH POINT OF THE
C***    AMBIGUITY SURFACE.  DO-LOOP 100 SEQUENCES THE SELECTION OF
C***    THE ROWS, DO-LOOP 110 SEQUENCES THE COLUMNS.  AS EACH ROW IS
C***    CALCULATE, IT IS WRITTEN TO FILE "ANS" FOR RECALL BY THE DATA
C***    FORMATTING ROUTINE "GRAPH", WHICH MUST BE RUN NEXT TO PLOT THE
C***    DATA.  THE MAGNITUDE IS CALCULATED IN THE USUAL MANNER.

        DO 100 I=1,64
           DO 110 J=1,64
              ANS(J)=SQRT(RANS(I,J)*RANS(I,J)+IANS(I,J)*IANS(I,J))
110        CONTINUE
           CALL WRBLK(6,I,ANS,1,IER)
100      CONTINUE

C***    THE FOLLOWING STATEMENTS CLOSE FILE "ANS", AND DISPLAYS THE
C***    STATUS OF THE CLOSURE: IER SHOULD BE 1 IF THE FILE WAS CLOSED
C***    CORRECTLY.

        CALL CLOSE(6,IER)
        TYPE 'FOR CLOSING FILE "ANS", IER=',IER
```

```
C***    THE NEXT STATEMENTS CALL SUBROUTINE "SAVDAT".  THE VARIABLES
C***    PASSED TO THE ROUTINE WILL BE WRITTEN TO FILE "PAMS"
C***    (PARAMETERS) AND LATER READ BY PROGRAM "GRAPH".

        CALL OVLOD(4,SAVE,1,IER)
        CALL SAVDAT(TEMP,TIME,WS,WIN,WF,NP,PW,PRT,RANGE,TESTP,FO,TESTF,
     *             PERCNT,PE1,PE2,FE1,FE2,NUM)

C***    THE NEXT STATEMENT CLOSES THE OVERLAY CAPABILITY

        CALL CLOSE(4,IER)

C***    FORMAT STATEMENTS

300     FORMAT(' THE CURRENT VALUE OF M IS ',I3)
310     FORMAT(' THE CURRENT VALUE OF N IS ',I3)
320     FORMAT(' VALUE OF M-N= ',I3)
330     FORMAT(' W NOW BEING EVALUATED = ',F16.6)

        END
```

```
C***********************************************************************
C*                                                                     *
C*      SUBROUTINE MSG1                                                *
C*                                                                     *
C*      WRITTEN BY: CAPT THOMAS L. GRIFFIN, JR.                        *
C*                                                                     *
C*      DATE: 25 SEP 85                                                *
C*                                                                     *
C*      THIS SUBROUTINE SIMPLY TYPES THE OPENING MESSAGES.  THE        *
C*      ROUTINE IS WRITTEN AS AN OVERLAY/ SUBROUTINE IN ORDER TO       *
C*      CONSERVE RAM MEMORY SPACE DURING EXECUTION OF "CONPULS".       *
C*                                                                     *
C*      THE RDOS COMMAND TO COMPILE THE SOURCE CODE IS: FORTRAN INTRO  *
C*                                                                     *
C***********************************************************************

      OVERLAY INTRO
      SUBROUTINE MSG1

      TYPE ' '
      TYPE '*******************    PROGRAM CONPULS    ******************'
      TYPE '  '
      TYPE 'THIS PROGRAM COMPUTES THE AMBIGUITY FUNCTION FOR THE '
      TYPE 'LINEAR, STEP FREQUENCY WAVEFORM.  THE PULSE ENVELOPE USED'
      TYPE 'IS RECTANGULAR, THE REPETITION TIME IS UNIFORM, THE '
      TYPE 'REPETITION RATE IS 2 OR 4 TIMES THE PULSE WIDTH, AND 2,4,'
      TYPE 'OR 8 PULSES MAY BE SELECTED.  A NUMBER OF OPTIONS'
      TYPE 'ARE AVAILABLE:  SIMULATE A CONSTANT CARRIER PULSE TRAIN,'
      TYPE 'INCLUDE RANDOM INITIAL PHASES IN 5 RANGES, INCLUDE RANDOM'
      TYPE 'CARRIER FREQUENCY DRIFTS OF UP TO 1.0 PERCENT OF THE '
      TYPE 'CARRIER FREQUENCY.  SEVERAL SUBROUTINES REQUIRE KEYBOARD'
      TYPE 'INPUT OF VARIABLES.  INSTRUCTIONS ARE GIVEN AS NEEDED.'
      TYPE ' '
      TYPE 'THIS PROGRAM REQUIRES PROGRAM "GRAPH" BE RUN TO FORMAT'
      TYPE 'THE THE DATA FOR PLOTTING.  PROGRAM "PLTTRNS" PLOTS THE'
      TYPE 'RESULTS.'
      TYPE ' '
      PAUSE 'ENTER ANY KEY TO CONTINUE'
      TYPE ' '
      TYPE '*** WHEN ASKED FOR KEYBOARD INPUT, ENSURE ONLY NUMBERS'
      TYPE '*** ARE ENTERED; OTHER CHARACTERS WILL CAUSE AN ERROR'
      TYPE '*** RESULTING IN A RETURN TO THE SYSTEM PROMPT.  NUMBERS'
      TYPE '*** ENTERED OTHER THAN THOSE REQUESTED WILL RESULT IN'
      TYPE '*** UNPREDICTABLE RESULTS.  IF AN ENTRY ERROR IS MADE,'
```

```
TYPE '*** ENTER A BACKSLASH (\) FOLLOWED BY THE CORRECT ENTRY.'
TYPE '*** THE CORRECTION MUST BE MADE BEFORE PRESSING "RETURN".'
TYPE '*** COMPLETE ALL ENTRIES BY PRESSING THE CARRIAGE RETURN.'
TYPE ' '
PAUSE 'ENTER ANY KEY TO CONTINUE'
TYPE ' '

RETURN
END
```

```
C***********************************************************************
C*                                                                     *
C*   SUBROUTINE INDAT (INPUT DATA)                                     *
C*                                                                     *
C*   WRITTEN BY: CAPT THOMAS L. GRIFFIN JR                             *
C*   DATE:  5 SEPTEMBER 1985                                           *
C*                                                                     *
C*   THE PURPOSE OF THIS ROUTINE IS TO ASSIGN (VIA KEYBOARD ENTRY)     *
C*   VALUES TO VARIABLES USED IN PROGRAM "CONPULS".  THOSE VARIABLES   *
C*   ARE:                                                              *
C*                                                                     *
C*      PW = WIDTH OF EACH PULSE IN THE WAVEFORM.  RANGE VARIES        *
C*           FROM X.0 TO 0.00000X SECONDS.                             *
C*      RT = PULSE REPETITION TIME.  LIMITED TO 2 OR 4 TIMES THE       *
C*           PULSEWIDTH.                                               *
C*      NP = NUMBER OF PULSES IN THE WAVEFROM: 2,4,8 OR 16.            *
C*      WS = STARTING VALUE OF DOPPLER FREQUENCY SHIFT IN CALCULATING  *
C*           THE AMBIGUITY FUNCTION: IN RADIANS.                       *
C*      WF = FINAL VALUE OF DOPPLER FREQUENCY SHIFT EVALUATED.         *
C*     ROW = AN INTEGER RELATED TO A SPECIFIC DOPPLER FREQUENCY FOR    *
C*           WHICH ALL VALUES OF THE AMBIGUITY SURFACE WILL BE         *
C*           TABULATED.                                                *
C*                                                                     *
C*   THE VALUES ASSIGNED TO THESE VARIABLES ARE USED TO DETERMINE THE  *
C*   FOLLOWING:                                                        *
C*                                                                     *
C*   RATIO = THE RATIO OF PRT TO PW.  USED AS A MULTIPLIER IN DETER-   *
C*           MINING THE POSITION OF THE TIME-DELAYED WAVEFORM.         *
C*    TIME = THE TIME INTERVAL BETWEEN SAMPLES OF EACH PULSE: USED IN  *
C*           "CONPULS" TO CALCUATE VALUES FOR TIME-VARYING PHASE TERMS.*
C*     NUM = NUMBER OF SAMPLES USED TO REPRESENT EACH PULSE: A POWER OF*
C*           TWO FROM 4 TO 64.                                         *
C*     WIN = FREQUENCY INCREMENT BETWEEN EACH DOPPLER SHIFT EVALUATED. *
C*    WCEN = CENTER OF THE DOPPLER FREQUENCY SHIFT RANGE.              *
C*   W(64) = AN ARRAY CONTAINING THE VALUE OF DOPPLER SHIFT WHICH WILL *
C*           BE USED FOR EACH ROW (CUT) OF THE AMBIGUITY FUNTION.      *
C*    SIZE = ASSIGNED VALUE OF 256: USED IN COMPUTING NUM.  256 IS THE *
C*           SIZE OF THE ARRAY IN WHICH EACH PULSE IS FOUND.           *
C*   LIMIT = A FUNCTION OF NP AND RATIO, USED IN COMPUTING NUM.        *
C*  CENTER = ASSIGNED VALUE OF 128, USED TO SET THE LEFT AND RIGHT     *
C*           EDGES OF EACH PULSE.                                      *
C*   START = ASSIGNED VALUE OF ZERO: REPRESENTS THE EXTREME LEFT SIDE  *
C*           OF THE ARRAY IN WHICH EACH PULSE IS FOUND.                *
C*    STOP = ASSIGNED VALUE OF 256: REPRESENTS THE EXTREME RIGHT SIDE. *
```

```
C*                                                                      *
C*   RATIONALE FOR THE RANGE AND LIMITS OF THESE VARIABLES IS DETAILED *
C*   IN CHAPTER II.  SOME RATIONALE IS GIVEN AS COMMENTS WITHIN THE     *
C*   ROUTINE.                                                           *
C*                                                                      *
C*   THE SUBROUTINE IS COMPILED AS AN OVERLAY IN ORDER TO CONSERVE      *
C*   MEMORY ALLOCATION SPACE WITHIN THE ECLIPSE COMPUTER.  THIS SOURCE *
C*   CODE HAS FILENAME "IN.FR".  REFERENCE DATA GENERAL'S               *
C*   FORTRAN 5 PROGRAMMMERS GUIDE, CHAPTER 15 FOR DETAILS OF            *
C*   HOW TO USE OVERLAYS.                                               *
C*                                                                      *
C*   THE RDOS COMMAND TO COMPILE THE SOURCE CODE IS: FORTRAN IN         *
C*                                                                      *
C***********************************************************************

      OVERLAY IN
      SUBROUTINE INDAT(PW,PRT,RATIO,NP,SIZE,LIMIT,NUM,CENTER,
     *          START,STOP,WS,WF,WIN,WCEN,W,TIME,ROW,TEST2)
      INTEGER CENTER,RATIO,NUM,NP,NN,SIZE,LIMIT,START,STOP,TST,ROW,
     *        TEST2,TEMP
      REAL    PW,PRT,W(64),INTERVAL,WS,WF,WIN,WCEN

      TYPE ' '
      TYPE '*****  SUBROUTINE INDAT (INPUT DATA)  *****'
      TYPE ' '
      TYPE 'THIS SUBROUTINE ASSIGNS VALUES VIA KEYBOARD INPUT TO'
      TYPE 'PROGRAM VARIABLES.  THEY ARE PULSEWIDTH (PW), PULSE REP-'
      TYPE 'ETITION TIME (PRT), THE NUMBER OF PULSES (NP), AND THE'
      TYPE 'DOPPLER FREQUENCY RANGE OVER WHICH THE AMBIGUITY FUNC-'
      TYPE 'TION IS COMPUTED.  SEVERAL OTHER VARIALBES ARE THEN COM-'
      TYPE 'PUTED INCLUDING THE MAXIMUM NUMBER OF SAMPLES POSSIBLE'
      TYPE 'TO REPRESENT EACH PULSE.'
      TYPE '  '
915   ACCEPT 'ENTER PULSEWIDTH: X.0 TO 0.00000X ',PW
910   TYPE ' '
      TYPE 'PULSE REPETITION TIME MUST BE 2 OR 4 TIMES PULSEWIDTH:'
      ACCEPT 'ENTER YOUR CHOICE: 2 OR 4. ',RATIO

C***  TEST TO SEE IF PRT IS 2 OR 4 TIMES PW; IF NOT START OVER.

      PRT=RATIO*PW
      IF (RATIO.LT.2) GO TO 900
      NN=1
      DO 10 J=1,2
```

```
         NN=NN*2
         IF (NN.EQ.RATIO)GO TO 15
10     CONTINUE
       GO TO 905

15     TYPE ' '
       TYPE 'THE NUMBER OF PULSES ALLOWED IN THE WAVEFORM IS '
       TYPE '2,4, OR 8.  THE NUMBER OF PULSES SIGNIFICANTLY EFFECTS'
       TYPE 'THE TIME REQUIRED FOR THE PROGRAM TO RUN.  TIME FOR TWO'
       TYPE 'PULSES IS APPROXIMATELY 4 MINUTES, 4 PULSES 17 MINUTES,'
       TYPE 'AND 8 PULSES APPROXIMATELY 68 MINUTES.'
       TYPE 'THEREFORE, WHEN PROMPTED TO SELECT THE NUMBER OF PULSES,'
       TYPE 'LET THESE APPROXIMATE TIMES TO RUN INFLUENCE YOUR CHOICE.'
       TYPE ' '
       TYPE 'ENTER THE NUMBER OF PULSES: 2,4,or 8 '
       ACCEPT NP

C***   CHECK TO SEE IF NP IS 2,4, or 8

       IF (NP.LT.2) GO TO 920
       NN=1
       DO 20 J=1,3
         NN=NN*2
         IF(NN.EQ.NP) GO TO 25
20     CONTINUE
       GO TO 925

25     SIZE=256
       LIMIT=2*NP*RATIO
       NUM=SIZE/LIMIT

C***   INTIIALIZE THE FOLLOWING VARIABLES

       CENTER=128
       TIME=PW/NUM
       START=0
       STOP=256

C***   THE NEXT SECTION SETS UP THE DOPPLER FREQUENCY SHIFT RANGE.
C***   RECOMMENDED VALUES ARE CALCULATED AND OFFERED AS NOMINAL INPUT
C***   FOR WS AND WF.  SINCE 2*NP-1 MAJOR PEAKS OCCUR IN THE AMBIGUITY
C***   SUFACE, AND SINCE A DOPPLER RANGE OF (2*NP-1)*1/PW*2 IS NEEDED
C***   TO CALCULATE AND VIEW THESE PEAKS, THEN +/- (2*NP-1)*1/PW IS
C***   NOMINALLY ASSIGNED TO WS AND WF RESPECTIVELY.
```

```
C***
C***    DO-LOOP 40 LOADS ARRAY "W".  SUCEEDING ELEMENTS OF THE ARRAY ARE
C***    1/64 OF THE TOTAL DOPPLER FREQUENCY RANGE GREATER THAN THE PAST.
C***    SIXTY-FOUR VALUES ARE THE MAXIMUN NUMBER POSSIBLE DUE TO THE LIM-
C***    ITATIONS OF THE PLOTTING PROGRAM "PLTTRNS".

        TYPE '*****'
        TYPE 'TO CALCULATE AND VIEW ALL MAJOR PEAKS OCCURRING IN THE'
        TYPE 'AMBIGUITY FUNCTION, A DOPPLER RANGE OF (2*NP-1)*1/PW*2'
        TYPE '(HERTZ) IS REQUIRED.  HOWEVER, TO VIEW AND COMPUTE VALUES'
        TYPE 'OF ONLY THE CENTRAL PEAK (AS IS OFTEN DESIRED), A MUCH'
        TYPE 'SMALLER RANGE OF DOPPLER IS NEEDED.'
        TYPE ' '
        TYPE 'THE STARTING AND FINAL VALUES OF THE DESIRED RANGE MUST'
        TYPE 'BE SELECTED NOW.  ENTER 1 TO SET VALUES SO THAT ALL'
        TYPE 'MAJOR PEAKS ARE DISPLAYED (THESE VALUES WILL BE DISPLAYED'
        TYPE 'IF SELECTED) OR ENTER 0 TO INPUT THE START AND STOP'
        ACCEPT 'VALUES MANUALLY: ', TST
        TYPE ' '

        IF(TST.EQ.1)GO TO 30

        ACCEPT 'ENTER STARTING VALUE OF DOPPLER SHIFT (W)- IN RADS ',WS
        ACCEPT 'ENTER FINAL VALUE OF W  ',WF
        TYPE ' '
        GO TO 35

30      WS=-(2*NP-1)*1/PW*2*3.1415927
        WF=(2*NP-1)*1/PW*2*3.1415927
        WRITE(10,360)WS,WF

35      WIN=(WF-WS)/64.0
        WCEN=(WF+WS)/2.0

        DO 40 I=1,64
          W(I)=WS+WIN*I
40      CONTINUE

C***    THE NEXT STATEMENT ACCEPTS KEYBOARD INPUT TO CONTROL PRINTING
C***    OF THE X ARRAY FOR A SPECIFIC DOPPLER SHIFT.  THE CONTROL
C***    PARAMETER FOR THIS DECISION IS "TEST2".  THE X ARRAY WILL, IN
C***    GENERAL, CONTAIN THE MAXIMUM VALUE OF THE AMBIGUITY SURFACE WHEN
C***    THE DOPPLER SHIFT IS ZERO.  THIS CODE AND SUBROUTINE "TST" WILL
C***    RESULT IN ALL 256 VALUES OF THE AMBIGUITY SURFACE (FOR THE
```

```
C***    DOPPLER SHIFT CHOSEN) TO BE PRINTED.

600     TYPE' '
        TYPE 'THE CAPABILITY EXISTS TO PRINTOUT THE VALUE OF EACH POINT'
        TYPE 'IN AN AMBIGUITY SURFACE FOR ANY OF THE 64 VALUES OF'
        TYPE 'DOPPLER SHIFT. DOING SO ALLOWS CLOSE EXAMINATION OF DATA.'
        TYPE 'THE DOPPLER VALUE IS COMPUTED AS FOLLOWS:'
        TYPE 'W=WS+(WS+WF)/64*N, WHERE N IS 1-64.  THE NUMBER N'
        TYPE '(1-64) IS ENTERED, AND THE CORRESPONDING DOPPLER SHIFT IS'
        TYPE 'DISPLAYED.  '
        ACCEPT '*** ENTER 1 TO SELECT OPTION, 0 OTHERWISE ***  ',TEST2

        IF(TEST2.EQ.0)GO TO 200
        TYPE ' '
        TYPE 'THE STARTING FREQUENCY ENTERED WAS ',WS
        TYPE 'THE FINAL FREQUENCY ENTERED WAS ',WF
        TYPE 'W = WS + (WF+WS)/64*N.  W = 0.0 IF WS = -WF'
        ACCEPT 'ENTER NUMBER (N) DESIRED: ',ROW
        TYPE ' '
        TYPE 'FOR THE NUMBER ',ROW,' THE DOPPLER SHIFT = ',W(ROW)
200     TYPE ' '
        ACCEPT 'ENTER 1 TO SELECT NEW NUMBER, 0 OTHERWISE',TEMP

        IF(TEMP.EQ.1)GO TO 600

        GO TO 100

C***    THESE LINES SHOULD ONLY BE EXECUTED IF THE INPUTS ARE WRONG ***

900     TYPE ' PRT MUST BE AT LEAST TWICE THE PW: ENTER 2 OR 4.'
        GO TO 910
905     TYPE ' THE RATIO OF PRT/PW WAS NOT 2 OR 4 TIMES PW'
        TYPE ' START OVER. ENTER PW: X.0 TO 0.00000X'
        GO TO 915
920     TYPE ' NUMBER OF PULSES MUST BE AT LEAST 2, START OVER:'
        GO TO 15
925     TYPE ' THE NUMBER OF PULSES ENTERED WAS NOT 2,4, OR 8.'
        GOTO 15

360     FORMAT(' STARTING DOPPLER SHIFT (IN RADS) =  ',F14.5,' FINAL SHI
     *FT = ',F14.5)

100     RETURN
        END
```

```
C************************************************************************
C*                                                                      *
C*       SUBROUTINE FERRORS (FREQUENCY ERRORS)                          *
C*                                                                      *
C*       WRITTEN BY: CAPT THOMAS L. GRIFFIN, JR.                        *
C*                                                                      *
C*       23 AUGUST 1985                                                 *
C*                                                                      *
C*       THE PURPOSE OF THIS SUBROUTINE IS TO CREATE TWO ARRAYS         *
C*       WHICH CONTAIN THE DRIFT IN THE DESIRED CARRIER FREQUENCY       *
C*       FOR EACH TRANSMITTED PULSE AND THE DRIFT IN THE LOCAL OSC      *
C*       ( TX AND RX DRIFT TYPES ).  THE DRIFT IS EITHER 1) ZERO,       *
C*       OR 2) A RANDOM PERCENTAGE LESS THAN 1.0 OF THE CARRIER         *
C*       FREQUENCY.  THE ROUTINE REQUIRES INPUT FROM THE KEYBOARD.      *
C*       INPUTS ARE: THE CONTROL VARIABLE (TESTF) WHICH CAUSES THE      *
C*       ERRORS TO ASSUME ALL ZERO VALUES (IF TESTF=0) OR TO ASSUME     *
C*       VALUES (IF TESTF=1); CARRIER FREQUENCY (FO).                   *
C*                                                                      *
C*       THE HEART OF THE ROUTINE IS A RANDOM NUMBER GENERATOR.         *
C*       IT GENERATES RANDOM NUMBERS BETWEEN 0 AND 999, THEN THE        *
C*       NUMBERS ARE THEN CONVERTED TO PERCENTAGES AND STORED IN        *
C*       ARRAYS FE1 AND FE2.                                            *
C*                                                                      *
C*       THE RDOS COMMAND TO COMPILE THE SOURCE CODE IS: FORTRAN FER    *
C*                                                                      *
C************************************************************************

      OVERLAY FER
      SUBROUTINE FERRORS(FO,FE1,FE2,NP,PW,TESTF,PERCNT)

      REAL FE1(8),FE2(8),FACTOR,FO,PW,PERCNT
      INTEGER TESTF,TESTPW,TESTP,TEMP

      TYPE ' '
      TYPE '*********  SUBROUTINE FERRROR  *********'
      TYPE ' '
      TYPE 'THIS SUBROUTINE CREATES RANDOM SHIFTS IN THE CARRIER'
      TYPE 'FREQUENCY (AS OPPOSED TO SHIFTS AS A FUNCTION OF TIME).'
      TYPE 'THE MAXIMUM SHIFT SIMULATED IS VARIABLE FROM 0.0 TO'
      TYPE '1.0%  IF THE OPTION TO INCLUDE CARRIER ERRORS IS'
      TYPE 'CHOSEN, THE CARRIER FREQUENCY MUST BE INPUT.  THIS FRE-'
      TYPE 'QUENCY SHOULD BE APPROXIMATELY 100 TIMES GREATER THAN'
      TYPE '1/PW, WHERE PW IS THE PULSEWIDTH USED.  THE SHIFTS ARE'
      TYPE 'PASSED TO PROGRAMS "CONPULS" AND "GRAPH".  GRAPH WILL'
```

```
      TYPE 'PRINT THE ERRORS IF DESIRED.  THIS ROUTINE WILL ALLOW'
      TYPE 'MULTIPLE ITERATIONS TO SELECT AND VIEW NEW RESULTS.'
      TYPE ' '
      ACCEPT 'ENTER 1 TO CREATE RANDOM FREQUENCY ERRORS OR 0 OTHERWISE
     * ',TESTF

      IF(TESTF.EQ.0)GO TO 10

C***  THE NEXT SECTION WILL DETERMINE THE CARRIER FREQUENCY (FO) USED
C***  IN "CONPULS".  THE OPTION TO SET FO EQUAL TO 1/PW IS OFFERED.

160   TYPE ' '
      TYPE 'THE CARRIER FREQUENCY IS NOMINALLY SET TO 100*1/PW. FOR'
      TYPE 'THE PW SELECTED, IT WOULD BE', 100/PW
      ACCEPT 'ENTER 1 TO CHANGE FO, 0 OTHERWISE ',TESTPW

      IF(TESTPW.EQ.1)GO TO 100
      FO=100/PW
      GO TO 110

100   TYPE ' '
      TYPE 'ENTER CARRIER FREQUENCY (100 UP TO 1,000,000,000) '
      ACCEPT FO

C***  THE NEXT SECTION WILL DETERMINE THE MAXIMUM FREQUENCY SHIFT
C***  SIMULATED.  IT IS NOMINALLY SET TO 1.0 PERCENT.

110   TYPE ' '
      TYPE 'THE MAXIMUM SHIFT IS NOMINALLY SET TO 1.0 PERCENT OF'
      TYPE 'THE CARRIER FREQUENCY.  ENTER 1 TO CHANGE, 0 OTHERWISE. '
      ACCEPT TESTP

      IF(TESTP.NE.0)GOTO 120

C***  IF THE 1.0 PERCENT OPTION IS CHOSEN, A FACTOR IS SET WHICH
C***  CONVERTS THE RANDOM NUMBERS GENERATED TO A PERCENTAGE BETWEEN
C***  0.0 AND 1.0.  A SIMILAR OPERATION IS PERFORMED AFTER KEYBOARD
C***  INPUT OF THE PERCENTAGE.  THE RANDOM NUMBERS GENERATED RANGE BE-
C***  TWEEN 2 AND 500.  NOTE: VARIABLE PERCNT IS ASSIGNED A VALUE OF
C***  1.0 IF THE 1.0 PERCENT OPTION IS SELECTED IN ORDER THAT IS VALUE
C***  MAY BE WRITTEN TO FILE "PAMS" LATER.  IT WILL BE USED WITHIN
C***  PROGRAM "GRAPH".

      PERCNT=1.0
```

```
      FACTOR=0.00001
      GO TO 130

120   TYPE ' '
      ACCEPT 'ENTER A PERCENTAGE FROM 0.0 TO 1.0 ',PERCNT
      FACTOR=PERCNT*0.00001

C***  THE NEXT SECTION INPUTS 4 SEEDS FOR THE RANDOM NUMBER GENERATOR.

130   TYPE ' '
      TYPE '4 SEEDS FOR THE RANDOM NUMBER GENERATOR ARE REQUIRED.'
      ACCEPT 'ENTER FIRST SEED (INTEGER FROM 1 TO 490 )',I1
      ACCEPT 'ENTER SECOND SEED (1-490)',I2
      ACCEPT 'ENTER THIRD SEED (1-490)',J1
      ACCEPT 'ENTER FOURTH SEED (1-490)',J2

C***  THE NEXT SECTION (TO THE FIRST "RETURN") CREATES THE RANDOM
C***  PERCENTAGES AND LOADS THEM INTO ARRAYS FE1 AND FE2.  THERE
C***  ARE ESSENTIALLY TWO IDENTICAL BLOCKS OF CODE; ONE TO FILL FE1,
C***  THE OTHER TO FILL FE2.  VARIABLES L AND M CONTROL THE MAGNITUDE
C***  OF THE RANDOM NUMBERS.  THE MAXIMUM NUMBER IS L+M-2, THE
C***  MINIMUM IS 2.  ONE RANDOM SHIFT IS NEEDED FOR EACH PULSE, SO THE
C***  LOOPS INCREMENT FROM 1 TO NP (THE NUMBER OF PULSES).  THE RANDOM
C***  NUMBER (TEMP) IS ACTUALLY CREATED AT LINES 30 AND 60.
C***  LINES 40 AND 70 ASSIGN THE NUMBER TO FE1 AND FE2 WHILE ALSO
C**   CONVERTING IT TO A PERCENTAGE.

      L=509
      M=491

      DO 15 K=1,NP
      IF(I1-L)20,25,25
25    I1=I1-L
20    IF(J1-M)30,35,35
35    J1=J1-M
30    TEMP=I1+J1
40    FE1(K)=TEMP*FACTOR
      I1=I1+I1
      J1=J1+J1

      IF(I2-L)50,55,55
55    I2=I2-L
50    IF(J2-M)60,65,65
65    J2=J2-M
```

```
60      TEMP=I2+J2
70      FE2(K)=TEMP*FACTOR
        I2=I2+I2
        J2=J2+J2
15      CONTINUE

C***    THE NEXT SECTION WILL ALLOW DISPLAYING THE PERCENTAGES WHICH ARE
C***    STORED IN THE TWO ARRAYS.  THE OPTION TO RESTART THE ROUTINE IS
C***    OFFERED IF IT IS DETERMINED THE PERCENTAGES GENERATED ARE NOT
C***    SATISFACTORY.

        TYPE ' '
        TYPE 'ENTER 1 TO DISPLAY THE RANDOM PERCENTAGES GENERATED,'
        ACCEPT 'OR 0 TO RETURN TO THE MAIN PROGRAM:  ',TEMP
        TYPE ' '
        IF(TEMP.EQ.0)GO TO 140
        DO 150 I=1,NP
           WRITE(10,300)I,FE1(I)*100,FE2(I)*100
150     CONTINUE
        WRITE(10,300)I,FE1(0)*100,FE2(0)*100
        TYPE ' '
        TYPE 'IF THESE VALUES ARE NOT ACCEPTABLE, ENTER 1 TO RESTART'
        TYPE 'THIS SUBROUTINE, AND CHOOSE NEW SEEDS.  IF THE VALUES'
        ACCEPT 'ARE SATISFACTORY, ENTER ZERO.   ',TEMP
        IF(TEMP.EQ.1)GOTO 160

140     RETURN

C***    DO-LOOP 80 IS EXECUTED ONLY IF THE OPTION FOR RANDOM FREQUENCY
C***    SHIFTS IS NOT TAKEN.  ARRAYS FE1 AND FE2 ARE SIMPLY FILLED WITH
C***    ZERO VALUES.  VARIABLES FO AND PERCNT ARE ASSIGNED ZERO VALUES
C***    FOR USE WITHIN SUBROUTINE "SAVDAT" AND PROGRAM "GRAPH".

10      DO 80 K=1,NP
           FE1(K)=0.0
           FE2(K)=0.0
80      CONTINUE
        FO=0.0
        PERCNT=0.0
        WRITE(10,300)I,FE1(0)*100,FE2(0)*100

        RETURN

300     FORMAT(' FOR PULSE # ',I2,' TX DRIFT= ',F7.4,' RX DIRFT= ',F7.4)
```

END

```
C***********************************************************************
C*                                                                     *
C*   SUBROUTINE PERRORS (PHASE ERRORS)                                 *
C*                                                                     *
C*   WRITTEN BY: CAPT THOMAS L. GRIFFIN, JR.                           *
C*                                                                     *
C*   DATE: 23 AUGUST 1985, MODIFIED 26 OCT 1985                        *
C*                                                                     *
C*   THE PURPOSE OF THIS SUBROUTINE IS TO CREATE TWO ARRAYS            *
C*   WHICH REPRESENT THE INITIAL PHASES OF EACH PULSE IN THE TRANS-    *
C*   MITTED WAVEFORM AND IN THE INPUT TO THE MATCHED FILTER IN THE     *
C*   RECEIVER.  THE RANGE OF INITIAL PHASE IS SELECTABLE IN FOUR       *
C*   STEPS: 20,45,90, OR 180 DEGREES.  EACH PULSE IN THE WAVEFORMS     *
C*   IS ASSIGNED A DIFFERENT INITIAL PHASE.                            *
C*                                                                     *
C*   THE HEART OF THE ROUTINE IS A PSUEDO-RANDOM NUMBER GENERATOR      *
C*   LISTED IN "MATHEMATICS AND COMPUTING WITH FORTRAN PROGRAMMING",   *
C*   BY DORN AND GREENBERG, JOHN WILEY AND SONS, INC., 1967, PP. 474-  *
C*   484.  IT GENERATES RANDOM NUMBERS BETWEEN 2 AND 358.  THESE       *
C*   NUMBERS ARE THEN CONVERTED TO RADIAN ANGLES AND STORED IN         *
C*   ARRAYS PE1 AND PE2.                                               *
C*                                                                     *
C*   THE SUBROUTINE IS COMPILED AS AN OVERLAY TO CONSERVE MEMORY       *
C*   ALLOCATION IN THE ECLIPSE COMPUTER.                               *
C*                                                                     *
C*   THE RDOS COMMAND TO COMPILE THE SOURCE CODE IS: FORTRAN PER       *
C*                                                                     *
C***********************************************************************

        OVERLAY PER
        SUBROUTINE PERRORS(PE1,PE2,NP,TESTP,RANGE)

        REAL PE1(8),PE2(8),FACTOR,RADS
        INTEGER TESTP,TEMP,RANGE,TEMP1

        TYPE ' '
        TYPE '**********   SUBROUTINE PERRORS ***********************'
        TYPE ' '
        TYPE 'THIS SUBROUTINE CREATES RANDOM, INITIAL PHASES OF THE'
        TYPE 'CARRIER FREQUENCY (AS OPPOSED TO SHIFTS AS A FUNCTION'
        TYPE 'OF TIME).  THE MAXIMUM PHASE SIMULATED IS VARIABLE IN '
        TYPE '4 RANGES: 20,45,90, AND 180 DEGREES.  THIS ROUTINE USES'
        TYPE 'KEYBOARD INPUT TO SELECT THE INITIAL PHASE OPTION, TO'
        TYPE 'SELECT THE RANGE, AND TO PROVIDE SEEDS FOR THE RANDOM'
```

```
      TYPE 'NUMBER GENERATOR'
      TYPE ' '
      TYPE 'IF YOU WANT TO INCLUDE INITIAL PHASES, '
      ACCEPT 'ENTER 1 NOW, OR 0 OTHERWISE  ',TESTP

      IF(TESTP.EQ.0)GO TO 10

160   TYPE ' '
      ACCEPT 'ENTER RANGE: 20,45,90, OR 180 DEGREES  ',RANGE

C***  THE NEXT SECTION REQUESTS AND ACCEPTS 4 SEEDS FOR THE RANDOM
C***  NUMBER GENERATOR.  THEIR RANGE IS ARBITARILY LIMITED TO 1-80.

      ACCEPT 'ENTER SEED (1-80)  ',I1
      ACCEPT 'ENTER SEED #2 (1-80)',I2
      ACCEPT 'ENTER SEED #3 (1-80)',J1
      ACCEPT 'ENTER SEED #4 (1-80)',J2

C***  THE NEXT BLOCK COMPUTES THE RANDOM PHASES AND FILLS ARRAYS PE1
C***  AND PE2.  THE SECTION CONTAINS TWO NEARLY IDENTICAL SECTIONS,
C***  WITHIN DO-LOOP 15.  THE LOOP FIRST CHECKS THAT THE I AND J SEEDS
C***  ARE LESS THAN L AND M RESPECTIVELY (IF NOT THEN THE VALUE OF L
C***  OR M IS SUBTRACTED FROM EACH).  A RANDOM NUMBER (TEMP) IS THEN
C***  COMPUTED BY ADDING THE I AND J VALUES.  THE LINES ASSIGNING
C***  TEMP TO THE PE-ARRAYS COMPARES TEMP TO 180 TO KEEP TEMP BELOW
C***  180 DEG.  THEN MULTIPLES BY FACTOR TO KEEP THE VALUES WITHIN THE
C***  SELECTED RANGE AND BY RADS TO CONVERT TO RADIANS.  FINALLY THE I
C***  AND J VARIABLES ARE DOUBLED TO PREPARE FOR THE NEXT LOOP-VALUE.
C***
C***  THE GREATEST NUMBER GENERATED IS EQUAL TO L+M-2=182;  THE
C***  SMALLEST IS 2.  L AND M MUST BE PRIME NUMBERS WHICH HAVE 2 AS A
C***  PRIMITIVE ROOT (REFERENCE CRC TABLES FOR LISTS OF THESE PRIME
C***  NUMBERS).

      FACTOR=RANGE/180.0
      RADS=0.0174533
      L=101
      M=83
      DO 15 K=1,NP
         IF(I1-L)20,25,25
25       I1=I1-L
20       IF(J1-M)30,35,35
35       J1=J1-M
30       TEMP=I1+J1
```

```
           IF(TEMP-182)40,45,45
45         TEMP=TEMP-2
40         PE1(K)=TEMP*FACTOR*RADS
           I1=I1+I1
           J1=J1+J1

           IF(I2-L)50,55,55
55         I2=I2-L
50         IF(J2-M)60,65,65
65         J2=J2-M
60         TEMP=I2+J2
           IF(TEMP-182)70,75,75
75         TEMP=TEMP-2
70         PE2(K)=TEMP*FACTOR*RADS
           I2=I2+I2
           J2=J2+J2
15      CONTINUE

C***    THE NEXT SECTION WILL ALLOW DISPLAY OF THE GENERATED PHASES.
C***    THEY ARE CONVERTED TO DEGREES FOR INTERPRETATION EASE.  THEN
C***    THE OPTION TO RESTART THE ROUTINE IS OFFERED IF IT IS DECIDED
C***    THAT THE PHASES GENERATED ARE NOT ACCEPTABLE.  THE "ANINT"
C***    STATEMENT ROUNDS TO THE NEAREST INTEGER.

        TYPE ' '
        TYPE 'ENTER 1 TO DISPLY RANDOM PHASES, OR 0 TO RETURN TO THE '
        ACCEPT 'MAIN PROGRAM: ',TEMP1
        IF(TEMP1.NE.1)GOTO 100
        TYPE ' '
        TEMP=1.0/RADS
        DO 105 I=1,NP
           WRITE(10,300)I,PE1(I)*TEMP,PE2(I)*TEMP
105     CONTINUE
        TYPE ' '
        TYPE 'IF THESE VALUES ARE NOT ACCEPTABLE, ENTER 1 TO RESTART'
        TYPE 'THIS ROUTINE, AND ENTER DIFFERENT SEEDS.  IF THE VALUES'
        ACCEPT 'ARE ACCEPTABLE ENTER 0  ',TEMP
        IF(TEMP.EQ.1)GOTO 160

100     RETURN


C***    DO-LOOP 80 FILLS THE TWO INITIAL PHASE ARRAYS WITH ZEROS.  IT IS
C***    EXECUTED ONLY IF THE OPTION TO INCLUDE INITIAL PHASES IS
```

```
C***     DECLINED.

10       DO 80 K=1,NP
            PE1(K)=0.0
            PE2(K)=0.0
80       CONTINUE
         RANGE=0
         RETURN

300      FORMAT(' PULSE NUMBER ',I2,' PE1 AND PE2 ERRORS ARE: ',F10.5,3X,
     *   F10.5)
         END
```

```
C************************************************************************
C*                                                                      *
C*     SUBROUTINE INITIAL                                               *
C*                                                                      *
C*     WRITTEN BY: CAPT THOMAS L. GRIFFIN JR.                           *
C*     DATE: 11 OCT 85                                                  *
C*                                                                      *
C*     THE PURPOSE OF THIS SUBROUTINE IS TO INITIALIZE SEVERAL VARI-    *
C*     ABLES AND DISPLAY SEVERAL MESSAGES.  THE ROUTINE WAS WRITTEN AS  *
C*     AN OVERLAY IN ORDER TO CONSERVE MEMORY SPACE WITHIN THE DATA     *
C*     GENERAL "ECLIPSE" COMPUTER.                                      *
C*                                                                      *
C*     THE RDOS COMMAND TO COMPILE THE SOURCE CODE IS: FORTRAN INIT     *
C*                                                                      *
C************************************************************************

      OVERLAY INIT
      SUBROUTINE INITIAL(NUM,NP,SCALE,PI,TEST1,FR,RANS,IANS,PW)

      INTEGER NUM,NP,TEST1
      REAL SCALE,PI,FR,RANS(64,64),IANS(64,64),PW


C***  THE NEXT BLOCK INITIALIZES PI AND THE FREQUENCY STEP SIZE (FR).
C***  FR IS NOMINALLY SET EQUAL TO 1/PW AS RECOMMENDED IN SEVERAL
C***  SOURCES.  HOWEVER, THE OPTION TO SET FR TO ZERO EXISTS IN ORDER
C***  TO COMPUTE THE AMBIGUITY FUNCTION OF A CONSTANT CARRIER FRE-
C***  QUENCY PULSETRAIN.  SCALE WILL BE USED TO NORMALIZE THE MAGNI-
C***  TUDE OF EACH ELEMENT IN AN ARRAY CALLED MAGNITUDE TO A VALUE OF
C***  1.0.  THE VALUE ASSIGNED TO SCALE COMES FROM THE FACT THAT IN A
C***  DISCRETE FOURIER TRANSFORM, THE MAXIMUM VALUE COMES FROM THE SUM
C***  (OVER THE NUMBER OF SAMPLES) OF EACH SAMPLE TIMES THE VALUE OF
C***  THE WAVEFORM AT THAT SAMPLE POINT.

      SCALE = 1.0/(NUM*NP)
      PI=3.1415927

      TYPE ' '
      TYPE 'A CONSTANT CARRIER FREQUENCY MAY BE SIMULATED BY SETTING'
      TYPE 'THE FREQUENCY STEP SIZE PARAMETER (FR) TO ZERO.  IF THIS'
      TYPE 'OPTION IS SELECTED, THE AMBIGUITY FUNCTION COMPUTED WILL'
      TYPE 'BE THAT OF A CONSTANT CARRIER PULSETRAIN. THIS OPTION CAN'
      TYPE 'BE USED TO VERIFY THE PROGRAM SINCE THAT PLOT IS WELL '
      TYPE 'KNOWN.  IF THE OPTION IS DECLINED, FR IS SET TO 1/PW.'
```

```
      TYPE ' '
      ACCEPT 'ENTER 1 TO SET FR TO ZERO: ENTER 0 OTHERWISE  ',TEST1
      IF(TEST1.EQ.1)FR=0.0
      IF(TEST1.EQ.0)FR=1.0/PW

C***  THESE NEXT LINES SIMPLY DELETE, CREATE, THEN OPEN A FILE NAMED
C***  "DATA" IN WHICH THE RESULTS OF THIS PROGRAM WILL BE STORED.  THE
C***  FILE WILL BE USED BY THE PLOTTING PROGRAM.  IER IS AN ERROR FLAG
C***  CREATED BY EACH SUBROUTINE CALLED, AND IT SHOULD BE "1" IF NO
C***  ERROR EXISTS.

      CALL DFILW("ANS",IER)
      TYPE 'FOR DFILW, IER=',IER
      CALL CFILW("ANS",3,65,IER)
      TYPE 'FOR CFILW ,IER=',IER
      CALL OPEN (6,"ANS",2,IER)
      TYPE 'FOR OPENING ANS, IER=',IER

      TYPE 'THE STATUS VARIABLE "IER" IN THE "STATUS" MESSAGES JUST'
      TYPE 'WRITTEN MUST EACH BE 1 OR ELSE A MAJOR ERROR HAS OCCURED'
      TYPE 'IN THE DELETION, CREATION, AND OPENING OF FILE "ANS".'
      TYPE 'NOTE: IER NOT EQUAL TO 1 FOR DFILW IS NOT FATAL TO THE'
      TYPE 'PROGRAM, BUT CFILW AND  OPENING "ANS" MUST BE 1 TO '
      TYPE 'CONTINUE.  IF NOT, ABORT THE PROGRAM (ENTER <CTRL A>),'
      TYPE 'AND INVESTIGATE THE PROBLEM.'
      TYPE ' '
      PAUSE 'ENTER ANY KEY TO CONTINUE'
      TYPE ' '

C***  DO LOOPS 10 AND 20 INSURE THE ANSWER ARRAYS ARE INITIALLY ZERO

      DO 10 I=1,64
         DO 20 J=1,64
            RANS(J,I)=0.0
            IANS(J,I)=0.0
20       CONTINUE
10    CONTINUE

      RETURN
      END
```

```
C************************************************************************
C*                                                                      *
C*  SUBROUTINE LDARYS (LOAD ARRAYS)                                     *
C*                                                                      *
C*  WRITTEN BY CAPT THOMAS L. GRIFFIN JR.                               *
C*                                                                      *
C*  25 AUGUST 1985                                                      *
C*                                                                      *
C*  THE PURPOSE OF THIS SUBROUTINE IS TO REPRESENT THE COMPLEX          *
C*  ENVELOPES OF TWO SINUSOIDAL PULSES IN ARRAYS SO THAT THE            *
C*  AMBIGUITY FUNCTION OF A RECTANGULAR PULSE TRAIN MAY BE PER-         *
C*  FORMED.  THIS REQUIRES TWO ARRAYS, ONE FOR A STATIONARY PULSE       *
C*  STARTING AT TIME ZERO BUT MULTIPLIED BY A DOPPLER TERM, AND ONE     *
C*  FOR A TIME REVERSED AND TIME DELAYED PULSE.  THIS ROUTINE IS CALLED*
C*  BY PROGRAM "CONPULS" FOR EACH VALUE OF DOPPLER SHIFT (64 SHIFTS),   *
C*  AND THE SEQUENCING OF THE 64 SHIFTS IS DONE N-SQUARED TIMES, WHERE *
C*  N IS THE NUMBER OF PULSES IN THE PULSETRAIN. THE MINOR DIFFERENCE   *
C*  IN CENTERING THE TIME=0 POSITION (ELEMENTS 129 VS 128) ALLOWS THE   *
C*  EVENTUAL CENTER OF THE CONVOLUTION OF THE TWO ARRAYS TO BE AT 128, *
C*  AND THERE TO BE 2*NUM-1 POINTS. COMMENTS WITHIN THE PROGRAM EXPLAIN*
C*  THE PROCEDURE.                                                      *
C*                                                                      *
C*  THE RDOS COMMAND TO COMPILE THE SOURCE CODE IS: FORTRAN LARS        *
C*                                                                      *
C************************************************************************

      OVERLAY LARS
      SUBROUTINE LDARYS(W,CENTER,NUM,Y,X,PW,PI,M,N,FR,FE1,FE2,FO,
     * RATIO,P,TIME)

      INTEGER LEFT,RIGHT,CENTER,NUM,RATIO
      REAL W(64),PW,TIME,PHASE1,PHASE2,PHASE3,PI,FR,FE1(8),FE2(8),
     *     FO
      COMPLEX X(256),Y(256),AMT1,AMT2,AMT3


C***  THIS NEXT SECTION OF CODE REPRESENTS THE STATIONARY PULSE.
C***  IT HAS AN AMPLITUDE OF 1 AND IS CENTERED IN THE MIDDLE OF
C***  ARRAY Y.  LEFT IS ITS LEFT EDGE, RIGHT IS ITS RIGHT EDGE.
C***  ALL OTHER VALUES IN THE ARRAY ARE ZERO.  TIME IS THE TIME
C***  INTERVAL BETWEEN SAMPLES.  PHASE1 AND PHASE2 ARE THE PHASE
C***  DELAYS OF EACH SAMPLE: PHASE1 IS DUE TO THE DOPPLER SHIFT,
C***  PHASE2 IS ACCOUNTS FOR THE STEP IN CARRIER FREQUENCY
```

```
      LEFT=129
      RIGHT=129+NUM-1
      DO 55 I=1,LEFT-1
         Y(I)=(0.0,0.0)
55    CONTINUE
      DO 60 I=LEFT,RIGHT
         Y(I)=(1.0,0.0)
60    CONTINUE
      DO 65 I=RIGHT+1,256
         Y(I)=(0.0,0.0)
65    CONTINUE

C***  DO-LOOP 90 APPLIES THE PHASE SHIFTS DUE TO DOPPLER AND THE
C***  CARRIER FREQUENCY SHIFTS TO THE ELEMENTS OF THE Y ARRAY (NON-
C***  ZERO VALUES).  AMT1= EXP(jwt); AMT2= EXP(j2pi(m-n)fr*t)

    DO 90 I=1,NUM-1
      PHASE1=W(P)*TIME*I
      PHASE2=2*PI*(M-N)*FR*TIME*I
      PHASE3=2*PI*(FE1(M+1)-FE2(N+1))*FO*TIME*I
      AMT1=CMPLX(0.0,PHASE1)
      AMT2=CMPLX(0.0,PHASE2)
      AMT3=CMPLX(0.0,PHASE3)
      Y(LEFT+I)=Y(LEFT+I)*CEXP(AMT1-AMT2-AMT3)
90  CONTINUE

C***  THE NEXT THREE LOOPS REPRESENT THE TIME REVERSED,DELAYED PULSE.
C***  THE CENTER OF THE PULSE IS NOT COMPUTED, ONLY THE LEFT AND RIGHT
C***  EDGES.  ELEMENTS OF THE ARRAY NOT REPRESENTING THE PULSE ARE SET
C***  TO ZERO.  NOTE: IF THE PULSE WERE NOT TIME-REVERSED (REQUIRED
C***  FOR STRICT COMPLIANCE WITH THE EXPRESSION BEING IMPLEMENTED) THE
C***  POSITIONS OF N AND M WOULD BE REVERSED.

      LEFT=CENTER-((N-M)*RATIO*NUM)-NUM+1
      RIGHT=CENTER-((N-M)*RATIO*NUM)
      DO 40 I=1,LEFT-1
          X(I)=(0.0,0.0)
40    CONTINUE
      DO 45 I=LEFT,RIGHT
         X(I)=(1.0,0.0)
45    CONTINUE
      DO 50 I=RIGHT+1,256
          X(I)=(0.0,0.0)
50    CONTINUE
```

```
RETURN
END
```

```
C*********************************************************************
C*                                                                   *
C*   SUBROUTINE TST1 (TEST 1)                                        *
C*                                                                   *
C*   WRITTEN BY: CAPT THOMAS L. GRIFFIN, JR.                         *
C*                                                                   *
C*   DATE: 26 AUGUST 85                                              *
C*                                                                   *
C*   THE PURPOSE OF THIS SUBROUTINE IS TO OBTAIN A PRINTOUT          *
C*   OF A SPECIFIC ROW IN THE AMBIGUITY SURFACE.  THE PRINTOUT       *
C*   WILL BE EXAMINED TO DETERMINE IF THE ROW IS SYMMETRICAL,        *
C*   AND TO FIND THE MAXIMUM AND THE -3DB VALUE WITHIN THE ROW.      *
C*                                                                   *
C*   THE ONLY VARIABLE PASSED TO THIS ROUTINE IS THE X ARRAY,        *
C*   WHICH CONTAINS THE 512 VALUES OF ROW 32 IN EACH AMBIGUITY       *
C*   SURFACE.  ROW 32 CORRESPONDS TO A DOPPLER SHIFT OF 0.0 HZ.      *
C*                                                                   *
C*   THE ROUTINE CREATES A TABLE OF 8 COLUMNS, EACH COLUMN CON-      *
C*   TAINS 64 VALUES.  A HEADING IS ALSO PRINTED OVER EACH COL.      *
C*                                                                   *
C*   THE RDOS COMMAND TO COMPILE THE SOURCE CODE IS: FORTRAN TST     *
C*                                                                   *
C*********************************************************************

      OVERLAY TST
      SUBROUTINE TST1(X,M,N,NP,ROW,TIME,Z,DOP)

      INTEGER TEMP,N,M,NP,ROW
      REAL MAX,HALF,TIME,DOP
      COMPLEX X(256),Z(256)

      TYPE 'INSIDE TST'

C***  DO-LOOP 5 IS EXECUTED ONLY FOR M=N=0:  THIS INITIALIZES
C***  THE Z-ARRAY.

      IF(M.NE.0.OR.N.NE.0)GO TO 7
      DO 5 I=1,256
         Z(I)=(0.0,0.0)
5     CONTINUE

C***  THIS LOOP WILL ADD TO ARRAY (Z) THE VALUES OF ARRAY (X). THE
C***  STATEMENT "CABS" TAKES THE MAGNITUDE OF X(I).
```

```
7       DO 10 I=1,256
           Z(I)=Z(I)+X(I)
10      CONTINUE

C***    THE NEXT SECTIONS SEARCH THE ARRAY (Z) FOR ITS LARGEST VALUE,
C***    MARKS THE VALUE'S SAMPLE NUMBER (MARK), AND COMPUTES THE TAU
C***    VALUE CORRESPONDING TO THE SAMPLE NUMBER (TAU).  THE SECTION IS
C***    ONLY EXECUTED IF M=N=(NP-1); THAT IS ONLY IF THE COMPUTATION OF
C***    ALL AMBIGUITY SURFACES OF THIS ROW HAS BEEN COMPLETED.  OTHERWISE,
C***    A RETURN TO PROGRAM "CONPULS" IS EXECUTED (LINE 100).

        TEMP=NP-1
        IF(M.NE.TEMP.OR.N.NE.TEMP)GO TO 100

        TYPE 'INSIDE PRINT LOOP'

        MAX=0.0
        DO 20 I=1,256
             IF(CABS(Z(I))-MAX)25,25,30
30           MAX=CABS(Z(I))
             TAU=TIME*(-128+I)
             MARK=I
25           CONTINUE
20      CONTINUE

C***    THIS SECTION FINDS THE -3DB VALUE CLOSEST TO THE MAXIMUM VALUE.
C***    THE VALUE IS STORED IN "DB3", ITS TAU POSITION COMPUTED (HALF),
C***    AND ITS SAMPLE POSITION MARKED (MARK1).

        HALF=0.0
        DO 32 I=MARK+1,256
           IF(ANINT(CABS(Z(I)))-ANINT(0.5*MAX))35,35,40
40         CONTINUE
32      CONTINUE
35      DB3=CABS(Z(I))
        HALF=TIME*(-128+I)
        MARK1=I

C***    THE WRITE STATEMENTS PROVIDE HARDCOPY RECORD OF THE PARAMETERS
C***    COMPUTED AND ALL VALUES WITHIN THE Z ARRAY.

        WRITE(12,310)ROW,DOP
        WRITE(12,315)MAX,TAU,MARK
        WRITE(12,320)DB3,HALF,MARK1
```

```
      WRITE(12,305)
      DO 45 I=1,32
         WRITE(12,300)CABS(Z(I)),CABS(Z(I+32)),CABS(Z(I+64)),
     *   CABS(Z(I+96)),CABS(Z(I+128)),CABS(Z(I+160)),
     *   CABS(Z(I+192)),CABS(Z(I+224))
45    CONTINUE

300   FORMAT(6X,F8.3,1X,F8.3,1X,F8.3,1X,F8.3,1X,F8.3,1X,F8.3,1X,F8.3,1X,
     *F8.3)

305   FORMAT('0',9X,'1-32   33-64    65-96   97-128  129-160  161-192
     *  193-224 225-256')

310   FORMAT('1',10X,'******  DATA FOR ROW ',I2,'-- W= ',F13.3,'        *
     ******')
315   FORMAT('0',9X,'MAXIMUM VALUE= ',F6.3,' AT TAU= ',F7.5,' (SAMPLE ',
     *I3,')')
320   FORMAT(9X,'-3DB VALUE= ',F6.3,' AT TAU= ',F7.5,' (SAMPLE ',I3,')')

100   RETURN
      END
```

```
C**********************************************************************
C*                                                                    *
C*    SUBROUTINE FOUREA                                               *
C*                                                                    *
C*    WRITTEN BY: CM RADER                                            *
C*    SOURCE:  "IEEE PROGRAMS FOR DIGITAL SIGNAL PROCESSSING", SECTION *
C*             1.1, (PP 1.1-1.5)                                      *
C*                                                                    *
C*    ADAPTED FOR THE GENERAL DATA 'ELCIPSE' BY CAPT DAVID KING, AFIT, *
C*    WPAFB OH.                                                       *
C*                                                                    *
C*    MODIFIED FOR USE BY PROGRAM CONPULS BY CAPT THOMAS L. GRIFFIN JR.*
C*                                                                    *
C*    THIS ROUTINE COMPUTES THE ONE DIMENSIONAL FAST FOURIER TRANSFORM *
C*    OF A COMPLEX INPUT ARRAY USING A RADIX 2 DECIMATION IN TIME     *
C*    ALGORITHM.  THE INPUT ARRAY IS FIRST BROKEN INTO ITS REAL AND   *
C*    IMAGINARY PARTS.  ADDITIONALLY, EACH ODD NUMBERED COMPONENT IS  *
C*    MULTIIPLIED BY -1.  THIS HAS THE EFFECT OF SHIFTING THE DC FREQ- *
C*    QUENCY COMPONENTS TO THE CENTER OF THE ARRAYS FROM THE LEFT EDGE.*
C*    THIS MODIFICATION WAS MADE IN ORDER THAT THE PLOT OF THE INVERSE *
C*    TRANSFORM WOULD HAVE ITS PEAK IN THE CENTER (AS IS CUSTOMARY    *
C*    FOR 3-D PLOTS: THE PURPOSE OF "CONPULS")                        *
C*                                                                    *
C*    MODICATIONS BY CAPT KING INCLUDE CLEANUP OF CODE AND CHECKING OF *
C*    INPUT PARAMETERS.                                               *
C*                                                                    *
C*    THE RDOS COMMAND TO COMPILE THE SOURCE CODE IS: FORTRAN FOUR    *
C*                                                                    *
C**********************************************************************

        SUBROUTINE FOUREA(AR,N,ISN)
        COMPLEX AR(512)
        DIMENSION XREAL(512),XIMAG(512)

C***    FOUREA - VER. 1.3 - 15 SEP 85


C***    THIS LOOP LOADS XREAL AND XIMAG

        DO 5 I=1,N
           XREAL(I)=REAL(AR(I))*((-1)**I)
           XIMAG(I)=AIMAG(AR(I))*((-1)**I)
5       CONTINUE
```

```
C***    CHECK ISN (1=INVERSE FFT, -1=FORWARD FFT)

        ISI=-1
        IF (ISN.EQ.1) ISI=1

C***    CHECK FOR POWER OF 2 UP TO 15

        IF (N.LT.2) GO TO 991
        NN=1
        DO 10 I=1,15
        NN=NN*2
        IF (NN.EQ.N) GO TO 20
10      CONTINUE
        GO TO 992
20      CONTINUE

        PI=4.*ATAN(1.)
        PI=FLOAT(ISI)*PI
        FN=N

C***    THIS SECTION BIT-REVERSES THE INPUT DATA

        J=1
        DO 80 I=1,N
        IF (I-J) 30,40,40
30      TEMPR=XREAL(J)
        TEMPI=XIMAG(J)
        XREAL(J)=XREAL(I)
        XIMAG(J)=XIMAG(I)
        XREAL(I)=TEMPR
        XIMAG(I)=TEMPI
40      M=N/2
50      IF (J-M) 70,70,60
60      J=J-M
        M=(M+1)/2
        GO TO 50
70      J=J+M
80      CONTINUE

C***    COMPUTE BUTTERFLIES

        MMAX=1
90      IF (MMAX-N) 100,130,130
```

```
100     ISTEP=2*MMAX
        DTHETA=PI/FLOAT(MMAX)
        DO 120 M=1,MMAX
        THETA=DTHETA*FLOAT(M-1)
        C=COS(THETA)
        S=SIN(THETA)
        DO 110 I=M,N,ISTEP
        J=I+MMAX
        TEMPR=C*XREAL(J)-S*XIMAG(J)
        TEMPI=C*XIMAG(J)+S*XREAL(J)
        XREAL(J)=XREAL(I)-TEMPR
        XIMAG(J)=XIMAG(I)-TEMPI
        XREAL(I)=XREAL(I)+TEMPR
        XIMAG(I)=XIMAG(I)+TEMPI
110     CONTINUE
120     CONTINUE
        MMAX=ISTEP
        GO TO 90
130     IF (ISI) 160,140,140

C***    FOR INVERSE TRANSFORM (ISI=1) - MULT BY 1/N

140     DO 150 I=1,N
        XREAL(I)=XREAL(I)/FN
        XIMAG(I)=XIMAG(I)/FN
150     CONTINUE
160     CONTINUE

C***    THIS LOOP LOADS THE INPUT COMPLEX ARRAY

        DO 170 I=1,N
           AR(I)=CMPLX(XREAL(I),XIMAG(I))
170     CONTINUE

        RETURN

C***    ERRORS

991     TYPE "<7>N = ",N
        STOP FOUREA01 - N MUST BE GREATER THAN 1
992     TYPE "<7>N = ",N
        STOP FOUREA02 - N MUST BE INTEGER POWER OF 2

        END
```

```
C***********************************************************************
C*                                                                     *
C*   SUBROUTINE SAVDAT (SAVE DATA)                                     *
C*                                                                     *
C*   WRITTEN BY:  CAPT THOMAS L. GRIFFIN, JR.                          *
C*                                                                     *
C*   18 SEPTEMBER 1985                                                 *
C*                                                                     *
C*   THE PURPOSE OF THIS ROUTINE IS TO WRITE SELECTED VARIABLES FROM   *
C*   WITHIN PROGRAM "CONPULS" TO FILE "PAMS" SO THAT THE VARIABLES MAY *
C*   LATER BE READ INTO PROGRAM "GRAPH".  GRAPH WILL PRODUCE A HARD    *
C*   COPY RECORD OF THE VARIABLES.                                     *
C*                                                                     *
C*   THE VARIABLES PASSED FROM CONPULS ARE:                            *
C*                                                                     *
C*   TEMP   = A REAL ARRAY (128) INTO WHICH THE FOLLOWING VARIABLES    *
C*            WILL BE WRITTEN.  TEMP IS THEN WRITTEN TO FILE "PAMS".   *
C*   TIME   = TIME INTERVAL BETWEEN SAMPLES OF EACH PULSE.             *
C*   WS     = STARTING DOPPLER FREQUENCY SHIFT EVALUATED IN COMPUTATION*
C*            OF THE AMBIGUITY FUNCTION.                               *
C*   WF     = FINAL DOPPLER FREQUENCY SHIFT EVALUATED.                 *
C*   WIN    = FREQUENCY INCREMENT BETWEEN SUCCESSIVE EVALUATIONS OF THE*
C*            FUNCTION.  WIN=(WS+WF)/64.                               *
C*   NP     = NUMBER OF PULSES IN THE PULSE TRAIN WAVEFORM.            *
C*   PW     = WIDTH OF EACH PULSE.                                     *
C*   PRT    = PULSE REPETITION TIME.                                   *
C*   RANGE  = MAXIMUM RANGE OVER WHICH INITIAL PHASE OF PULSES MAY     *
C*            VARY.                                                    *
C*   TESTP  = AN INTEGER (1 OR 0) WHICH REPRESENTS THE OPTION TO       *
C*            INCLUDE INITIAL PHASES (1) OR NOT (0).                   *
C*   FO     = THE CARRIER FREQUENCY.  NOMINALLY EQUAL TO 1/PW.         *
C*   TESTF  = AN INTEGER (1 OR 0) WHICH REPRESENTS THE OPTION TO       *
C*            INCLUDE CARRIER FREQUENCY DRIFTS IN EACH PULSE (1) OR    *
C*            NOT (0).                                                 *
C*   PERCNT = THE MAXIMUM PERCENTAGE OF CARRIER FREQUENCY DRIFT.       *
C*   PE1    = A REAL ARRAY (8) WHICH CONTAINS THE INITIAL PHASES       *
C*            APPLIED TO THE "TRANSMITTED" PULSE TRAIN.                *
C*   PE2    = A REAL ARRAY (8) WHICH CONTAINS THE INITIAL PHASES       *
C*            APPLIED TO THE "RECEIVED" PULSE TRAIN.                   *
C*   FE1    = A REAL ARRAY (8) WHICH CONTAINS THE CARRIER FREQUENCY    *
C*            DRIFTS APPLIED TO THE "TRANSMITTED" PULSE TRAIN.         *
C*   FE2    = A REAL ARRAY (8) WHICH CONTAINS THE CARRIER FREQUENCY    *
C*            DRIFTS APPLIED TO THE "RECEIVED" PULSE TRAIN.            *
C*                                                                     *
```

```
C*   THE SUBROUTINE IS COMPILED AS AN OVERLAY.  THIS SOURCECODE IS     *
C*   IN FILE "SAVE.FR".                                                *
C*                                                                     *
C*   THE RDOS COMMAND TO COMPILE THE SOUCE CODE IS: FORTRAN SAVE       *
C*                                                                     *
C***********************************************************************

        OVERLAY SAVE
        SUBROUTINE SAVDAT(TEMP,TIME,WS,WIN,WF,NP,PW,PRT,RANGE,TESTP,FO,
     *                    TESTF,PERCNT,PE1,PE2,FE1,FE2,NUM)

        INTEGER NP,RANGE,TESTP,TESTF,NUM
        REAL TEMP(128),TIME,WS,WIN,WF,PW,PRT,FO,PERCNT,PE1(8),PE2(8),
     *       FE1(8),FE2(8)


C***    THE FOLLOWING STATEMENTS FIRST DELETES OLD FILE "PAMS", THEN
C***    CREATES A NEW FILE "PAMS".  IN THE CFILW STATEMENT, THE '3'
C***    MAKES PAMS A CONTIGOUS FILE, AND THE '2' SPECIFIES THE FILE TO
C***    BE TWO BLOCKS OF 256 BYTES EACH (SEE DATA GENERAL'S MANUAL
C***    "FORTRAN 5 PROGRAMMER'S GUIDE (RDOS)", PART TWO, CHAPTER 5 FOR
C***    DETAILS).  THE FILE "PAMS" IS THEN OPENED.  THE '2' ALLOWS IT
C***    TO BE WRITTEN TO OR READ FROM.  PAMS WILL BE PASSED THE
C***    VARIABLES DESCRIBED ABOVE.

        CALL DFILW("PAMS",IER)
        TYPE 'FOR DFILW PAMS, IER=',IER
        CALL CFILW("PAMS",3,1,IER)
        TYPE 'FOR CFILW PAMS,IER=',IER
        CALL OPEN (6,"PAMS",2,IER)
        TYPE 'FOR OPENING PAMS, IER=',IER

C***    THE FOLLOWING STATEMENTS ASSIGN VALUES TO ELEMENTS OF THE ARRAY
C***    "TEMP".  TEMP WILL BE WRITTEN INTO FILE "PAMS".

        TEMP(1)=TIME
        TEMP(2)=WS
        TEMP(3)=WIN
        TEMP(4)=WF
        TEMP(5)=NP
        TEMP(6)=PW
        TEMP(7)=PRT
        TEMP(8)=RANGE
        TEMP(9)=TESTP
```

```
      TEMP(10)=FO
      TEMP(11)=TESTF
      TEMP(12)=PERCNT
      TEMP(13)=NUM

      DO 10 I=1,NP
         J=I+13
         K=J+NP
         L=J+2*NP
         M=J+3*NP
         TEMP(J)=PE1(I)
         TEMP(K)=PE2(I)
         TEMP(L)=FE1(I)
         TEMP(M)=FE2(I)
10    CONTINUE


C***  THE FOLLOWING STATEMENTS FIRST WRITES ARRAY "TEMP" INTO FILE
C***  "PAMS", DISPLAYS A STATUS MESSAGE, THEN CLOSES FILE "PAMS" AND
C***  DISPLAYS A STATUS MESSAGE.  REFERENCE DATA GENERAL'S MANUAL
C***  DESCRIBED ABOVE, CHAPTER 6 FOR DETAILS OF THE WRITE BLOCK
C***  (WRBLK) STATEMENT.

      CALL WRBLK(6,0,TEMP,1,IER)
      TYPE 'FOR WRITING TEMP, IER=',IER
      TYPE ' '
      CALL CLOSE(6,IER)
      TYPE 'FOR CLOSING PAMS,IER=',IER
      TYPE ' '
      TYPE 'THE "IER" VALUES SHOULD BE 1; IF NOT "GRAPH WILL NOT'
      TYPE 'RUN CORRECTLY.'
      TYPE ' '

C***  THE NEXT FEW STATEMENTS ARE END OF PROGRAM COMMENTS

      TYPE 'THE MAGNITUDE OF THE AMBIGUITY SURFACE IS CALCULATED AND'
      TYPE 'WRITTEN TO FILE "ANS".  AT THE SYSTEM PROMPT, TYPE GRAPH'
      TYPE 'TO FORMAT THE DATA FOR PLOTTING AND TABULATING.'

      RETURN
      END
```

Appendix B

<u>Program Listings for GRAPH</u>

This appendix contains the listings for program GRAPH and its subroutines: TABULATE and VARS.

The source code was written, compiled, and linked on the Data General Eclipse S/250 computer located in the AFIT Signal Processing Laboratory. The operating system was General Data's RDOS, version 7.41. The source code was compiled by Data General's Fortran 5 compiler, RDLR, version 6.16 and linked by linker version 7.40.

The source, object, and executable code was located on disk DP4 under directory EGRIFFIN. The entire directory has been written to magnetic tape for archival under the care of Mr. Dan Zambon, technician for the Eclipse computer system.

```
C***********************************************************************
C*                                                                     *
C*      PROGRAM GRAPH                                                  *
C*                                                                     *
C*      WRITTEN BY: CAPT THOMAS L GRIFFIN, JR.                         *
C*                                                                     *
C*      DATE: 6 SEPT 85                                                *
C*                                                                     *
C*      THE PROGRAM HAS TWO PURPOSES.  THE FIRST IS TO FORMAT THE DATA *
C*      FILE "ANS" SO THAT THE DATA CAN BE PLOTTED BY PROGRAM "PLTTRNS".*
C*      THE SECOND IS TO OPTIONALLY TABUALATE AND PRINT THE DATA.      *
C*                                                                     *
C*      THAT SECTION OF THE PROGRAM WHICH FORMATS THE DATA IS A MODIFI- *
C*      CATION OF CODE WRITTEN BY CAPT JOHN REED FOR A MASTER'S THESIS *
C*      IN 1982.                                                       *
C*                                                                     *
C*      THE RDOS COMMAND TO COMPILE THE SOURCE CODE IS: FORTRAN GRAPH  *
C*                                                                     *
C***********************************************************************

      INTEGER TAB1,GRAPH(64,64),TAB,ROW,DB3,MARK,MARK1,COLUMN,MAX,
     *        REPEAT,HC,NP,TESTF,TESTP,NUM,RANGE
      REAL TEMP(128),ANS(64,64),TIME,WS,WF,WIN,TAU,HALF,PW,PRT,
     *     FO,PERCNT,PE1(8),PE2(8),FE1(8),FE2(8)
      COMPLEX STORE(64)

      TYPE ' '
      TYPE '*********  PROGRAM GRAPH ******************************'
      TYPE ' '
      TYPE 'THIS PROGRAM HAS TWO PURPOSES: 1. TO FORMAT THE DATA FILE'
      TYPE '"ANS" (CREATED BY PROGRAM "CONPULS") FOR PLOTTING BY '
      TYPE 'PROGRAM "PLTTRANS"  2. TO OPTIONALLY TABULATE AND PRINT'
      TYPE 'A PERMANENT RECORD OF THE DATA.  THE PROGRAM PROMPTS THE'
      TYPE 'OPERATOR FOR RESPONSES AS IT COMES TO OPTIONS.'
      TYPE ' '
      TYPE '****  NOTE:  PROGRAM "CONPULS" MUST HAVE BEEN PREVIOUSLY'
      TYPE '****         RUN TO CREATE THE DATA ARRAYS REQUIRED OF '
      TYPE '****         THIS PROGRAM.  IF UNSURE, EXIT THIS PROGRAM'
      TYPE '****         AND CHECK'
      TYPE ' '
      TYPE '*********  CAUTION ***************************************'
      TYPE ' WHEN ASKED FOR KEYBOARD INPUT, ENSURE ONLY NUMBERS'
      TYPE ' ARE ENTERED; OTHER CHARACTERS WILL CAUSE AN ERROR'
      TYPE ' RESULTING IN A RETURN TO THE SYSTEM PROMPT.  NUMBERS'
```

```
      TYPE ' ENTERED OTHER THAN THOSE REQUESTED WILL RESULT IN'
      TYPE ' UNPREDICTABLE RESULTS.  IF AN ENTRY ERROR IS MADE,'
      TYPE ' ENTER A BACKSLASH (\) FOLLOWED BY THE CORRECT ENTRY.'
      TYPE ' COMPLETE ALL ENTRIES BY PRESSING THE CARRIAGE RETURN.'
      TYPE ' '
      PAUSE 'ENTER ANY KEY TO CONTINUE'
      TYPE ' '
      TYPE 'SEVERAL "STATUS" MESSAGES WILL APPEAR FIRST AS DATA IS '
      TYPE 'READ AND PROCESSED.  EACH "IER" VALUE MUST BE 1 OR AN '
      TYPE 'ERROR HAS OCCURED.  IF AN ERROR OCCURS, FIRST ENSURE THAT'
      TYPE 'FILES "ANS" AND "PAMS" EXIST.  IF NOT, RUN PROGRAM'
      TYPE '"CONPULS".  IF SO, THEN AN UNKNOWN ERROR EXISTS, AND DE-'
      TYPE 'GUGGING BEYOND THE SCOPE OF THIS MESSAGE IS REQUIRED.'
      PAUSE 'ENTER ANY KEY TO CONTINUE'

C***  READ DATA FILE "ANS" INTO ARRAY ANS.  THE FILE WAS CREATED BY
C***  PROGRAM "CONPULS".

      CALL OPEN(6,"ANS",2,IER)
      TYPE 'FOR OPENING ANS, IER=',IER
      DO 10 I=1,64
         CALL RDBLK(6,I,TEMP,1,IER)
         DO 20 J=1,64
            ANS(I,J)=TEMP(J)
20       CONTINUE
10    CONTINUE
      TYPE 'FOR READING LAST BLOCK OF "ANS", IER=',IER
      CALL CLOSE(6,IER)
      TYPE 'FOR CLOSING ANS, IER=',IER

C***  DO-LOOP 30 ENSURES ALL ELEMENTS OF ARRAY "TEMP" ARE INTIALLY ZERO

      DO 30 I=1,128
         TEMP(I)=0.0
30    CONTINUE

C***  READ DATA FILE "PAMS" INTO ARRAY TEMP, THEN ASSIGN VALUES TO
C***  ALL THE VARIABLES.  FILE "PAMS" WAS CREATED BY PROGRAM "CONPULS",
C***  SUBROUTINE "SAVDAT".  THE SIZE OF "PAMS" IS ONE BLOCK: 256 BYTES.
C***  SINCE ARRAY "TEMP" IS TO CONTAIN REAL DATA, AND SINCE IT IS TO BE
C***  DESTINATION ON FILE "PAMS", AND SINCE REAL DATA OCCUPIES 2 BYTES,
C***  THEN THE MAXIMUM SIZE "TEMP" CAN IS 128.

      CALL OPEN (6,"PAMS",2,IER)
```

```
      TYPE 'FOR OPENING PAMS, IER=',IER
      CALL RDBLK(6,0,TEMP,1,IER)
      TYPE 'FOR READING "PAMS" INTO "TEMP", IER=',IER
      CALL CLOSE(6,IER)
      TYPE 'FOR CLOSING FILE "PAMS", IER=',IER

      TIME=TEMP(1)
      WS=TEMP(2)
      WIN=TEMP(3)
      WF=TEMP(4)
      NP=TEMP(5)
      PW=TEMP(6)
      PRT=TEMP(7)
      RANGE=TEMP(8)
      TESTP=TEMP(9)
      FO=TEMP(10)
      TESTF=TEMP(11)
      PERCNT=TEMP(12)
      NUM=TEMP(13)
      DEGCON=57.29578
      DO 40 I=1,NP
         J=I+13
         K=J+NP
         L=J+NP+NP
         M=J+NP+NP+NP
         PE1(I)=TEMP(J)*DEGCON
         PE2(I)=TEMP(K)*DEGCON
         FE1(I)=TEMP(L)
         FE2(I)=TEMP(M)
40    CONTINUE

      TYPE ' '
      TYPE 'ALL "IER" VALUES ABOVE MUST BE 1 TO OBTAIN CORRECT'
      TYPE 'RESULTS.  IF SO ENTER ANY KEY TO CONTINUE, OR CONTROL A'
      TYPE 'TO ABORT THIS PROGRAM:'
      PAUSE
      TYPE '***** THE FOLLOWING WERE READ FROM FILE "PAMS" *****'
      TYPE ' '
      TYPE 'TIME=',TIME,' WS=',WS,' WIN=',WIN,' WF=',WF
      TYPE 'NP=',NP,' PW=',PW,' PRT=',PRT,' NUM= ',NUM,' RANGE=',RANGE
      TYPE 'TESTP=',TESTP,' FO=',FO,' TESTF=',TESTF,' PERCNT=',PERCNT
      DO 50 I=1,NP
         TYPE I,'=',PE1(I),PE2(I),FE1(I)*100,FE2(I)*100
50    CONTINUE
```

```
      TYPE ' '
      TYPE 'THESE VALUES SHOULD CORRESPOND TO THOSE ENTERED IN'
      TYPE 'OR CALCULATED BY PROGRAM "CONPULS"'
      TYPE ' '
      PAUSE ' ENTER ANY KEY TO CONTINUE'
      TYPE ' '
      TYPE '...REFORMATTING MAGNITUDES...WAIT...'

C***  THIS NEXT BLOCK (THRU LINE #105) SCALES THE MAGNITUDE DATA
C***  TO INTEGER VALUES BETWEEN 0 AND 200.  THE SCALED VALUES ARE THEN
C***  PLACED INTO ARRAY "GRAPH".  INTEGER VALUES ARE REQUIRED FOR THE
C***  PLOTTING PROGRAM "PLTTRNS".  THE KEY TO UNDERSTANDING THIS
C***  SECTION IS TO RECALL THAT THE MAXIMUM MAGNITUDE IN ARRAY "ANS"
C***  IS 1.0 (SINCE EACH REAL AND IMAGINARY VALUE WAS MULTIPIED BY THE
C***  INVERSE OF THE MAXIMUM VALUE POSSIBLE (VARIABLE "SCALE") IN PROGRAM
C***  "CONPULS".  NOW THE MAGNITUDE VALUE IS BEING SCALED UP BETWEEN
C***  1 AND 200 (INTEGER VALUES).

        DO 102 I=1,64
          DO 105 J=1,64
            G=ANS(I,J)
            K=0
            DO 110 L=1,200
              R=0.005*L
              IF (G.GE.R) K=L
110         CONTINUE
            GRAPH(I,J)=K
105       CONTINUE
102     CONTINUE

C***  THE NEXT BLOCK (THRU LINE "TYPE 'FOR ...") RETRIVES THE DATA
C***  FROM ARRAY GRAPH ONE ROW AT A TIME, PUTS IT IN ARAY "STORE"
C***  (THE PLOTTING PROGRAM "PLTTRNS" REQUIRES DATA TO BE COMPLEX)
C***  AND WRITES IT TO FILE "DATA".  FILE "DATA" IS FIRST
C***  DELETED, THEN CREATED, AND FINALLY OPENED.

      CALL DFILW("DATA",IER)
      CALL CFILW("DATA",2,IER)
      TYPE 'FOR CREATING DATA, IER=',IER
      CALL OPEN(6,"DATA",2,IER)
      TYPE 'FOR OPENING DATA, IER=',IER

      DO 115 I=1,64
```

```
      DO 120 J=1,64
         STORE(J)=CMPLX(GRAPH(I,J),0.0)
120   CONTINUE
      CALL WRBLK(6,I,STORE,1,IER)
115   CONTINUE
      CALL CLOSE(6,IER)
      TYPE 'FOR CLOSING DATA, IER=',IER
      TYPE ' '
      TYPE 'AGAIN, ALL "IER" VALUES ABOVE MUST BE ONE TO CONTINUE'
      TYPE 'CORRECTLY.  ENTER ANY KEY TO CONTINUE, CTRL A TO ABORT'
      PAUSE

C***  THE NEXT SECTION WILL ASK IF THE OPERATOR REQUESTS TABULATION
C***  OF DATA.  IF SO, TOWARDS THE END OF THIS PROGRAM, A CALL TO SUB-
C***  ROUTINE "TABULATE" IS MADE.  IT IS CAPABLE OF TABULATING ROWS OR
C***  COLUMNS OF DATA.

      TYPE ' '
      TYPE 'THE CAPABILITY EXITS TO TABULATE DATA BY ROWS OR COLUMNS.'
      TYPE 'FOR ROWS, THE MAGNITUDE DATA TABULATED CORRESPONDS TO'
      TYPE 'VALUES OF TAU (DELAY) FOR A GIVEN VALUE OF W (DOPPLER).'
      TYPE 'FOR COLUMNS, THE DATA IS MAGNITUDE FOR VALUES OF DOPPLER'
      TYPE 'FOR A GIVEN VALUE OF DELAY.  DATA VALUES ARE INTEGERS'
      TYPE 'BETWEEN 1 AND 200.'
      TYPE ' '
      TYPE 'ENTER 1 TO TABULATE A ROW OR COLUMN, 0 OTHERWISE. '
      ACCEPT TAB1

C***  THE NEXT STATEMENT CALLS "TABULATE" ONLY IF THE OPERATOR
C***  ENTERED A "1" WHEN ASKED IF IT WAS DESIRED TO TABULATE
C***  DATA.

      IF(TAB1.EQ.1)CALL TABULATE(GRAPH,TIME,WS,WIN,WF,NP,PW,PRT,
     * RANGE,TESTP,FO,TESTF,PERCNT,PE1,PE2,FE1,FE2,NUM)

C***  THE NEXT FEW TYPE STATEMENTS ARE END OF PROGRAM COMMENTS TO
C***  THE OPERATOR, AND ARE NOT REQUIRED FOR CORRECT OPERATION.

      TYPE ' '
      TYPE 'RESULTS HAVE BEEN TRANSFERRED TO FILE "DATA".'
      TYPE 'ENTER THE FOLLOWING COMMAND ON THE TEKTRONIX 4010'
      TYPE 'TO PLOT THE DATA: "PLTTRNS DATA/I 64/N"'
      TYPE 'FOLLOW THE PROMPTS TO OBTAIN A 3-D PLOT'
      TYPE 'FOR PLOTTING PURPOSES, TAU IS CENTERED AT 0.0'
```

```
      WRITE(10,350) TIME*(-512)
      WRITE(10,355) TIME*512

C***  FORMAT STATEMENTS

350   FORMAT(' FOR PLOTTING PURPOSES, TAU STARTS AT ',F10.6)
355   FORMAT(' FOR PLOTTING PURPOSES, TAU STOPS AT ',F10.6)

      END
```

```
C***********************************************************************
C*                                                                     *
C*   SUBROUTINE TABULATE                                               *
C*                                                                     *
C*   WRITTEN BY: Capt THOMAS L. GRIFFIN, JR.                           *
C*                                                                     *
C*   27 AUGUST 1985                                                    *
C*                                                                     *
C*   THE PURPOSE OF THIS ROUTINE IS TO CREATE A TABLE OF THE VALUES    *
C*   COMPUTED IN THE PROGRAM "CONPULS".  THIS ROUTINE WILL TABULATE    *
C*   THE VALUES STORED IN ARRAY ANSWER (ANS), WHICH IS 64X64, INTEGER  *
C*   VALUES.  THE ROUTINE WILL TABULATE ONLY ONE ROW OR ONE COLUMN AT  *
C*   A TIME.  SEE CHAPTER 3 AND COMMENNT LINES BELOW FOR MORE DETAILS. *
C*                                                                     *
C*   THE RDOS COMMAND TO COMPILE THE SOURCE CODE IS: FORTRAN TABULATE  *
C*                                                                     *
C***********************************************************************

      SUBROUTINE TABULATE(GRAPH,TIME,WS,WIN,WF,NP,PW,PRT,RANGE,TESTP,
     *     FO,TESTF,PERCNT,PE1,PE2,FE1,FE2,NUM)

      INTEGER GRAPH(64,64),TAB1,ROW,DB3,MARK,MARK1,COLUMN,MAX,REPEAT,
     *        HC,NP,RANGE,TESTP,TESTF,NUM
      REAL TAU,HALF,TIME,WS,WIN,WF,PW,PRT,FO,PERCNT,PE1(8),PE2(8),
     *      FE1(8),FE2(8)

      TYPE ' '
      TYPE '***********  SUBROUTINE TABULATE  ***********************'
      TYPE ' '
      TYPE 'THIS ROUTINE WILL ALLOW MULTIPLE TABULATIONS OF MAGNITUDE'
      TYPE 'DATA BY ROW OR COLUMN.  AFTER EACH TABULATION, YOU WILL'
      TYPE 'BE ASKED IF YOU WANT TO TABULATE ANOTHER ROW OR COLUMN.'
      TYPE 'THE OPTION TO OBTAIN HARD COPY WILL ALSO BE PRESENTED'
      TYPE 'EACH TIME.  NOTE: THE TABLE IS ALWAYS PRESENTED TO THE'
      TYPE 'SCREEN.'
5     TYPE ' '
      TYPE 'ENTER 1 TO TABULATE A ROW, 0 FOR A COLUMN'
      ACCEPT TAB1
      TYPE ' '
      TYPE 'ENTER 1 TO OBTAIN A PRINTOUT, 0 OTHERWISE'
      ACCEPT HC

      IF(TAB1.NE.1) GO TO 10
```

```
      TYPE ' '
      TYPE 'TABULATE WHICH ROW (1-64): ENTER ROW NUMBER'
      ACCEPT ROW
      MAX=0
      DO 20 I=1,64
         IF(GRAPH(ROW,I)-MAX)25,25,30
30       MAX=GRAPH(ROW,I)
         TAU=TIME*(-256+8*I)
         MARK=I
25       CONTINUE
20    CONTINUE

      HALF=0
      DO 32 I=MARK+1,64
         IF(GRAPH(ROW,I)-0.5*MAX)35,35,40
40       CONTINUE
32    CONTINUE
35    DB3=GRAPH(ROW,I)
      HALF=TIME*(-256+8*I)
      MARK1=I

      WRITE(10,300)ROW,WS+WIN*ROW
      IF(HC.EQ.1)WRITE(12,300)ROW,WS+WIN*ROW
300   FORMAT('1',9X,'ROW #= ',I2,6X,'(DOPPLER SHIFT= ',F10.3,')')
      IF(HC.EQ.1)WRITE(12,310)
310   FORMAT('  ')

      WRITE(10,320)MAX,TAU,MARK
      IF(HC.EQ.1)WRITE(12,320)MAX,TAU,MARK
320   FORMAT(10X,'MAXIMUM VALUE = 'I3,' AT TAU= ',F10.6,'  (SAMPLE #',
     *       I3,')')


      WRITE(10,330)DB3,HALF,MARK1
      IF(HC.EQ.1)WRITE(12,330)DB3,HALF,MARK1
330   FORMAT(11X,'-3dB VALUE = ',I3,' AT TAU= ',F10.6,' (SAMPLE #',
     *      I2,')')
      TYPE ' '
      WRITE(10,340)
      IF(HC.EQ.1)WRITE(12,340)
340   FORMAT('0',10X,'TAU',8X,'MAGNITUDE',6X,'TAU',8X,'MAGNITUDE')
      DO 45 I=1,32
         TAU=TIME*(-256+8*I)
         TAU1=TIME*(-256+8*(I+32))
```

```
      WRITE(10,350)TAU,GRAPH(ROW,I),TAU1,GRAPH(ROW,I+32)
      IF(HC.EQ.1)WRITE(12,350)TAU,GRAPH(ROW,I),TAU1,GRAPH(ROW,I+32)
350   FORMAT(9X,F12.9,4X,I3,7X,F12.9,4X,I3)
45    CONTINUE

      CALL VARS(NP,PW,PRT,NUM,TIME,WS,WF,WIN,TESTP,RANGE,TESTF,FO,
     *     PERCNT,PE1,PE2,FE1,FE2,HC)

      TYPE 'TABULATE ANOTHER ROW OR COLUMN? IF SO,'
      TYPE 'ENTER 1 NOW, 0 OTHERWISE.'
      ACCEPT REPEAT
      IF(REPEAT.EQ.1)GO TO 5
      RETURN

C***  THE NEXT SECTION OF CODE TABULATES COLUMN DATA

10    TYPE ' '
      TYPE 'TABULATE WHICH COLUMN (1-64): ENTER COLUMN NUMBER'
      ACCEPT COLUMN

      MAX=0
      DO 50 I=1,64
         IF(GRAPH(I,COLUMN)-MAX)55,55,60
60       MAX=GRAPH(I,COLUMN)
         DOPPLER=WS+WIN*I
         MARK=I
55       CONTINUE
50    CONTINUE

      HALF=0
      DO 65 I=MARK+1,64
         IF(GRAPH(I,COLUMN)-0.5*MAX)70,70,75
75       CONTINUE
65    CONTINUE
70       DB3=GRAPH(I,COLUMN)
         HALF=WS+I*WIN
         MARK1=I

      TAU=TIME*(-256+8*COLUMN)

      WRITE(10,400)COLUMN,TAU
      IF(HC.EQ.1)WRITE(12,400)COLUMN,TAU
400   FORMAT('1',9X,'COLUMN #',I2,8X,'(TAU = ',F10.6,')')
```

```
      WRITE(10,410)MAX,DOPPLER,MARK
      IF(HC.EQ.1)WRITE(12,410)MAX,DOPPLER,MARK
410   FORMAT('0',9X,'MAX VALUE = ',I3,' AT W=',F13.3,
     *  '   (SAMPLE #',I2,')')

      WRITE(10,420)DB3,HALF,MARK1
      IF(HC.EQ.1)WRITE(12,420)DB3,HALF,MARK1
420   FORMAT(9X,' -3dB VALUE= ',I3,' AT W=',F10.3,4X,
     *  '(SAMPLE #',I2,')')

      WRITE(10,430)
      IF(HC.EQ.1)WRITE(12,430)
430   FORMAT('0',9X,'DOPPLER SHIFT',5X,'MAGNITUDE',5X,'DOPPLER SHIFT',
     *  5X,'MAGNITUDE')

      DO 80 I=1,32
         WRITE(10,440)WS+WIN*I,GRAPH(I,COLUMN),WS+WIN*(I+32),
     *           GRAPH(I+32,COLUMN)
         IF(HC.EQ.1)WRITE(12,440)WS+WIN*I,GRAPH(I,COLUMN),
     *     WS+WIN*(I+32),GRAPH(I+32,COLUMN)
440      FORMAT( 9X,F10.3,11X,I3,8X,F10.3,11X,I3)
80    CONTINUE

      CALL VARS(NP,PW,PRT,NUM,TIME,WS,WF,WIN,TESTP,RANGE,TESTF,FO,
     *     PERCNT,PE1,PE2,FE1,FE2,HC)

      TYPE 'TABULATE ANOTHER COLUMN OR ROW? IF SO,'
      TYPE 'ENTER 1 NOW, 0 OTHERWISE'
      ACCEPT REPEAT
      IF(REPEAT.EQ.1)GOTO 5
      RETURN
      END
```

```
C***********************************************************************
C*                                                                     *
C*                                                                     *
C*      SUBROUTINE VARS (VARIABLES)                                    *
C*                                                                     *
C*      WRITTEN BY:  CAPT THOMAS L. GRIFFIN, JR.                       *
C*                                                                     *
C*      19 SEPTEMBER 1985                                              *
C*                                                                     *
C*                                                                     *
C*      THE PURPOSE OF THIS ROUTINE IS TO WRITE TO THE SCREEN AND TO THE*
C*      PRINTER VARIABLES USED IN COMPUTING THE AMBIGUITY FUNCTION OF A *
C*      LINEAR STEP-FREQUENCY PULSE TRAIN WAVEFORM.  THE VARIABLES ARE  *
C*      ASSIGNED VALUES WITHIN SUBROUTINES OF PROGRAM "CONPULS" AND ARE *
C*      THEN WRITTEN TO FILE "PAMS".  PROGRAM "GRAPH" READS "PAMS" AND  *
C*      PASSES THE VARIABLES TO SUBROUTINE "TABULATE".  TABULATE THEN   *
C*      PASSES THEM TO THIS ROUTINE.                                   *
C*                                                                     *
C*      "TABULATE" CALLS THIS ROUTINE FROM TWO LOCATIONS, BUT IN EACH   *
C*      CASE THIS ROUTINE EXECUTES THE SAME.  THE ROUTINE WAS WRITTEN   *
C*      IN THE INTEREST OF MODULARITY.                                 *
C*                                                                     *
C*      THE PARAMETERS PASSED FROM "TABULATE" ARE ALL LISTED IN THE    *
C*      SUBROUTINE DECLARATION STATEMENT.  THIS ROUTINE RETURNS NO     *
C*      VARIABLES.  DESCRIPTIONS OF THE VARAIBLES CAN BE FOUND IN      *
C*      PROGRAM "CONPULS", BUT SIMPLE DESCRIPTIONS CAN BE FOUND HERE   *
C*      WITHIN THE VARIOUS WRITE STATMENTS.                            *
C*                                                                     *
C*      THE RDOS COMMAND TO COMPILE THE SOURCE CODE IS: FORTRAN VARS   *
C*                                                                     *
C***********************************************************************

      SUBROUTINE VARS(NP,PW,PRT,NUM,TIME,WS,WF,WIN,TESTP,RANGE,TESTF,
     *FO,PERCNT,PE1,PE2,FE1,FE2,HC)

      INTEGER NP,NUM,TESTP,RANGE,TESTF,HC

      REAL PW,PRT,TIME,WS,WF,WIN,FO,PERCNT,PE1(8),PE2(8),FE1(8),
     *FE2(8)

C***  ALL OF THE WRITE STATEMENTS BELOW PRODUCE TEXT WHICH FOLLOWS
C***  OTHER TEXT WRITTEN BY SUBROUTINE "TABULATE".  THE FORMATTING
C***  IS PURELY ARBITARY, BUT ATTEMPTS TO MINIMIZE THE SPACE REQUIRED
```

```
C***  TO PRINT.  THE WRITE STATEMENTS THEMSELVES ARE STRAIGHT FORWARD.
C***  LOGICAL DEVICE #10 ON THE ECLIPSE (FOR WHICH THIS ROUTINE WAS
C***  WRITTEN) IS THE TERMINAL SCREEN, #12 IS THE LINE PRINTER.  VARIABLE
C***  "HC" (HARD COPY) DETERMINES WHETHER OR NOT THE WRITE-TO-PRINTER
C***  STATEMENTS ARE EXECUTED.

      WRITE(10,500)NP,PW
      IF(HC.EQ.1)WRITE(12,500)NP,PW
500   FORMAT('0',10X,'WAVEFORM PARAMETERS: # OF PULSES= ',I2,4X,'PULSEWI
     *DTH= ',F8.6)

      WRITE(10,510)PRT,NUM,TIME
      IF(HC.EQ.1)WRITE(12,510)PRT,NUM,TIME
510   FORMAT(12X,'PRT= ',F8.6,4X,'# OF SAMPLES= ',I2,4X,'SPACING= ',
     *       F11.9)

      WRITE(10,520)WS
      IF(HC.EQ.1)WRITE(12,520)WS
520   FORMAT('0',10X,'DOPPLER PARAMETERS: 1ST FREQ= ',F12.1)

      WRITE(10,530)WF,WIN
      IF(HC.EQ.1)WRITE(12,530)WF,WIN
530   FORMAT(12X,'LAST FREQ= ',F11.1,4X,'FREQ INCREMENTS= ',F15.5)

      IF(TESTP.EQ.0)WRITE(10,540)
      IF(TESTP.EQ.0.AND.HC.EQ.1)WRITE(12,540)
540   FORMAT('0',10X,'INITIAL PHASES:  OPTION DECLINED (ALL SET TO ZERO)
     *')

      IF(TESTP.EQ.1)WRITE(10,550)RANGE
      IF(TESTP.EQ.1.AND.HC.EQ.1)WRITE(12,550)RANGE
550   FORMAT('0',10X,'INITIAL PHASES:  RANGE= ',I3,' DEGREES')

      IF(TESTF.EQ.0)WRITE(10,560)
      IF(TESTF.EQ.0.AND.HC.EQ.1)WRITE(12,560)
560   FORMAT('0',10X,'CARRIER DRIFT:  OPTION DECLINED (ALL SET TO ZERO)
     *')

      IF(TESTF.EQ.1)WRITE(10,570)FO,PERCNT
      IF(TESTF.EQ.1.AND.HC.EQ.1)WRITE(12,570)FO,PERCNT
570   FORMAT('0',10X,'CARRIER DRIFT: FO= ',F12.1,' HZ  MAX PCNT DRIFT= '
     *,F5.3)

      IF(TESTF.NE.1.AND.TESTP.NE.1)GO TO 10
```

```
      WRITE(10,580)
      IF(HC.EQ.1)WRITE(12,580)
580   FORMAT('0',10X,'PULSE',6X,'PHASE (DEGREES)',10X,'CARRIER DRIFT (HZ
     *)')
      WRITE(10,590)
      IF(HC.EQ.1)WRITE(12,590)
590   FORMAT(' ',12X,'#',6X,'TX',15X,'RX',10X,'TX',16X,'RX')

      DO 20 I=1,NP
         WRITE(10,600)I,PE1(I),PE2(I),FE1(I)*FO,FE2(I)*FO
         IF(HC.EQ.1)WRITE(12,600)I,PE1(I),PE2(I),FE1(I)*FO,FE2(I)*FO
20    CONTINUE
600   FORMAT(' ',12X,I2,3X,F6.1,10X,F6.1,4X,F10.1,6X,F10.1)

10    RETURN
      END
```

Appendix C

Program Listings for GRAPH2

This appendix contains the listings for program GRAPH2 and its subroutines: TAB2 and VARS.

The source code was written, compiled, and linked on the Data General Eclipse S/250 computer located in the AFIT Signal Processing Laboratory. The operating system was General Data's RDOS, version 7.41. The source code was compiled by Data General's Fortran 5 compiler, RDLR, version 6.16 and linked by linker version 7.40.

The source, object, and executable code was located on disk DP4 under directory EGRIFFIN. The entire directory has been written to magnetic tape for archival under the care of Mr. Dan Zambon, technician for the Eclipse computer system.

```
C***********************************************************************
C*                                                                     *
C*   PROGRAM GRAPH2                                                    *
C*                                                                     *
C*   WRITTEN BY: CAPT THOMAS L GRIFFIN, JR.                            *
C*                                                                     *
C*   DATE: 26 OCT 85                                                   *
C*                                                                     *
C*   THE PROGRAM HAS ONE PURPOSE:  TO TABUALATE AND PRINT THE DATA     *
C*   STORED IN ARRAY "ANS" (CREATED BY PROGRAM "CONPULS"). THE PROGRAM *
C*   PROMPTS THE OPERATOR FOR RESPONSES AS IT COMES TO OPTIONS.        *
C*                                                                     *
C*   THE RDOS COMMAND TO COMPILE THE SOURCE CODE IS: FORTRAN GRAPH2    *
C*                                                                     *
C*   THE RDOS COMMAN TO LINK GRAPH2 IS:                                *
C*   RLDR GRAPH2 TAB2 VARS @FLIB@                                      *
C*                                                                     *
C***********************************************************************

      INTEGER TAB1,TAB,ROW,DB3,MARK,MARK1,COLUMN,MAX,
     *         REPEAT,HC,NP,TESTF,TESTP,NUM,RANGE
      REAL TEMP(128),ANS(64,64),TIME,WS,WF,WIN,TAU,HALF,PW,PRT,
     *      FO,PERCNT,PE1(16),PE2(16),FE1(16),FE2(16)

      TYPE ' '
      TYPE '*********  PROGRAM GRAPH2 ********************************'
      TYPE ' '
      TYPE 'THIS PROGRAM HAS ONE PURPOSE:  TO TABULATE AND PRINT'
      TYPE 'A PERMANENT RECORD OF THE DATA STORED IN ARRAY "ANS".'
      TYPE 'THE PROGRAM PROMPTS THE OPERATOR FOR RESPONSES AS IT'
      TYPE 'COMES TO OPTIONS.'
      TYPE ' '
      TYPE '****  NOTE:  PROGRAM "CONPULS" MUST HAVE BEEN PREVIOUSLY'
      TYPE '****         RUN TO CREATE THE DATA ARRAYS REQUIRED OF '
      TYPE '****         THIS PROGRAM.  IF UNSURE, EXIT THIS PROGRAM'
      TYPE '****         AND CHECK'
      TYPE ' '
      TYPE '*********  CAUTION ****************************************'
      TYPE ' WHEN ASKED FOR KEYBOARD INPUT, ENSURE ONLY NUMBERS'
      TYPE ' ARE ENTERED; OTHER CHARACTERS WILL CAUSE AN ERROR'
      TYPE ' RESULTING IN A RETURN TO THE SYSTEM PROMPT.  NUMBERS'
      TYPE ' ENTERED OTHER THAN THOSE REQUESTED WILL RESULT IN'
      TYPE ' UNPREDICTABLE RESULTS.  IF AN ENTRY ERROR IS MADE,'
      TYPE ' ENTER A BACKSLASH (\) FOLLOWED BY THE CORRECT ENTRY.'
```

```
      TYPE ' COMPLETE ALL ENTRIES BY PRESSING THE CARRIAGE RETURN.'
      TYPE ' '
      PAUSE 'ENTER ANY KEY TO CONTINUE'
      TYPE ' '
      TYPE 'SEVERAL "STATUS" MESSAGES WILL APPEAR FIRST AS DATA IS '
      TYPE 'READ AND PROCESSED.  EACH "IER" VALUE MUST BE 1 OR AN '
      TYPE 'ERROR HAS OCCURED.  IF AN ERROR OCCURS, FIRST ENSURE THAT'
      TYPE 'FILES "ANS" AND "PAMS" EXIST.  IF NOT, RUN PROGRAM'
      TYPE '"CONPULS".  IF SO, THEN AN UNKNOWN ERROR EXISTS, AND DE-'
      TYPE 'GUGGING BEYOND THE SCOPE OF THIS MESSAGE IS REQUIRED.'
      PAUSE 'ENTER ANY KEY TO CONTINUE'

C***  READ DATA FILE "ANS" INTO ARRAY ANS.  THE FILE WAS CREATED BY
C***  PROGRAM "CONPULS".

      CALL OPEN(6,"ANS",2,IER)
      TYPE 'FOR OPENING ANS, IER=',IER
      DO 10 I=1,64
         CALL RDBLK(6,I,TEMP,1,IER)
         DO 20 J=1,64
            ANS(I,J)=TEMP(J)
20       CONTINUE
10    CONTINUE
      TYPE 'FOR READING LAST BLOCK OF "ANS", IER=',IER
      CALL CLOSE(6,IER)
      TYPE 'FOR CLOSING ANS, IER=',IER

C***  DO-LOOP 30 ENSURES ALL ELEMENTS OF ARRAY "TEMP" ARE INTIALLY ZERO

      DO 30 I=1,128
         TEMP(I)=0.0
30    CONTINUE

C***  READ DATA FILE "PAMS" INTO ARRAY TEMP, THEN ASSIGN VALUES TO
C***  ALL THE VARIABLES.  FILE "PAMS" WAS CREATED BY PROGRAM "CONPULS",
C***  SUBROUTINE "SAVDAT".  THE SIZE OF "PAMS" IS ONE BLOCK: 256 BYTES.
C***  SINCE ARRAY "TEMP" IS TO CONTAIN REAL DATA, AND SINCE IT IS TO BE
C***  DESTINATION ON FILE "PAMS", AND SINCE REAL DATA OCCUPIES 2 BYTES,
C***  THEN THE MAXIMUM SIZE "TEMP" CAN BE IS 128.

      CALL OPEN (6,"PAMS",2,IER)
      TYPE 'FOR OPENING PAMS, IER=',IER
      CALL RDBLK(6,0,TEMP,1,IER)
      TYPE 'FOR READING "PAMS" INTO "TEMP", IER=',IER
```

```
      CALL CLOSE(6,IER)
      TYPE 'FOR CLOSING FILE "PAMS", IER=',IER

      TIME=TEMP(1)
      WS=TEMP(2)
      WIN=TEMP(3)
      WF=TEMP(4)
      NP=TEMP(5)
      PW=TEMP(6)
      PRT=TEMP(7)
      RANGE=TEMP(8)
      TESTP=TEMP(9)
      FO=TEMP(10)
      TESTF=TEMP(11)
      PERCNT=TEMP(12)
      NUM=TEMP(13)
      DEGCON=57.295779951
      DO 40 I=1,NP
         J=I+13
         K=J+NP
         L=J+NP+NP
         M=J+NP+NP+NP
         PE1(I)=TEMP(J)*DEGCON
         PE2(I)=TEMP(K)*DEGCON
         FE1(I)=TEMP(L)
         FE2(I)=TEMP(M)
40    CONTINUE

      TYPE 'ALL "IER" VALUES ABOVE MUST BE 1 TO OBTAIN CORRECT'
      TYPE 'RESULTS.  IF SO ENTER ANY KEY TO CONTINUE, OR CONTROL A'
      TYPE 'TO ABORT THIS PROGRAM:'
      PAUSE
      TYPE '***** THE FOLLOWING WERE READ FROM FILE "PAMS" *****'
      TYPE ' '
      TYPE 'TIME=',TIME,' WS=',WS,' WIN=',WIN,' WF=',WF
      TYPE 'NP=',NP,' PW=',PW,' PRT=',PRT,' RANGE=',RANGE
      TYPE 'TESTP=',TESTP,' FO=',FO,' TESTF=',TESTF,' PERCNT=',PERCNT
      DO 50 I=1,NP
         TYPE I,'=',PE1(I),PE2(I),FE1(I),FE2(I)
50    CONTINUE

      TYPE ' '
      TYPE 'VIEW RESULTS, THEN ENTER ANY KEY TO CONTINUE'
      PAUSE
```

```
C***    THIS NEXT BLOCK (THRU LINE #105) SCALES THE MAGNITUDE DATA
C***    TO VALUES BETWEEN 0.0 AND NUM.  THE SCALED VALUES ARE THEN
C***    PLACED INTO ARRAY "ANS".  THE KEY TO UNDERSTANDING THIS
C***    SECTION IS TO RECALL THAT THE MAXIMUM MAGNITUDE IN ARRAY "ANS"
C***    IS 1.0 (SINCE EACH REAL AND IMAGINARY VALUE WAS MULTIPIED BY THE
C***    INVERSE OF THE MAXIMUM VALUE POSSIBLE (VARIABLE "SCALE") IN PROGRAM
C***    "CONPULS".  NOW THE MAGNITUDE VALUE IS BEING SCALED UP BETWEEN
C***    0.0 AND SCALE (INTEGER VALUES).

          DO 102 I=1,64
            DO 105 J=1,64
              ANS(I,J)=ANS(I,J)*NP*NUM
105         CONTINUE
102       CONTINUE

        TYPE ' '
        TYPE 'THE CAPABILITY EXITS TO TABULATE DATA BY ROWS OR COLUMNS.'
        TYPE 'FOR ROWS, THE MAGNITUDE DATA TABULATED CORRESPONDS TO'
        TYPE 'VALUES OF TAU (DELAY) FOR A GIVEN VALUE OF W (DOPPLER).'
        TYPE 'FOR COLUMNS, THE DATA IS MAGNITUDE FOR VALUES OF DOPPLER'
        TYPE 'FOR A GIVEN VALUE OF DELAY.  DATA VALUES ARE REAL NUMBERS'
        TYPE 'BETWEEN 1 AND ',NUM*NP
        TYPE ' '
        TYPE 'ENTER 1 TO TABULATE A ROW OR COLUMN, 0 OTHERWISE. '
        ACCEPT TAB1

C***    THE NEXT STATEMENT CALLS "TABULATE" ONLY IF THE OPERATOR
C***    ENTERED A "1" WHEN ASKED IF IT WAS DESIRED TO TABULATE
C***    DATA.

        IF(TAB1 EQ.1)CALL TAB2(ANS,TIME,WS,WIN,WF,NP,PW,PRT,
     *  RANGE,TESTP,FO,TESTF,PERCNT,PE1,PE2,FE1,FE2,NUM)

C***    THE NEXT FEW TYPE STATEMENTS ARE END OF PROGRAM COMMENTS TO
C***    THE OPERATOR, AND ARE NOT REQUIRED FOR CORRECT OPERATION.

        TYPE ' '
        TYPE 'RUN PROGRAM "GRAPH" TO FORMAT DATA FOR PLOTTING'
        TYPE ' '

        END
```

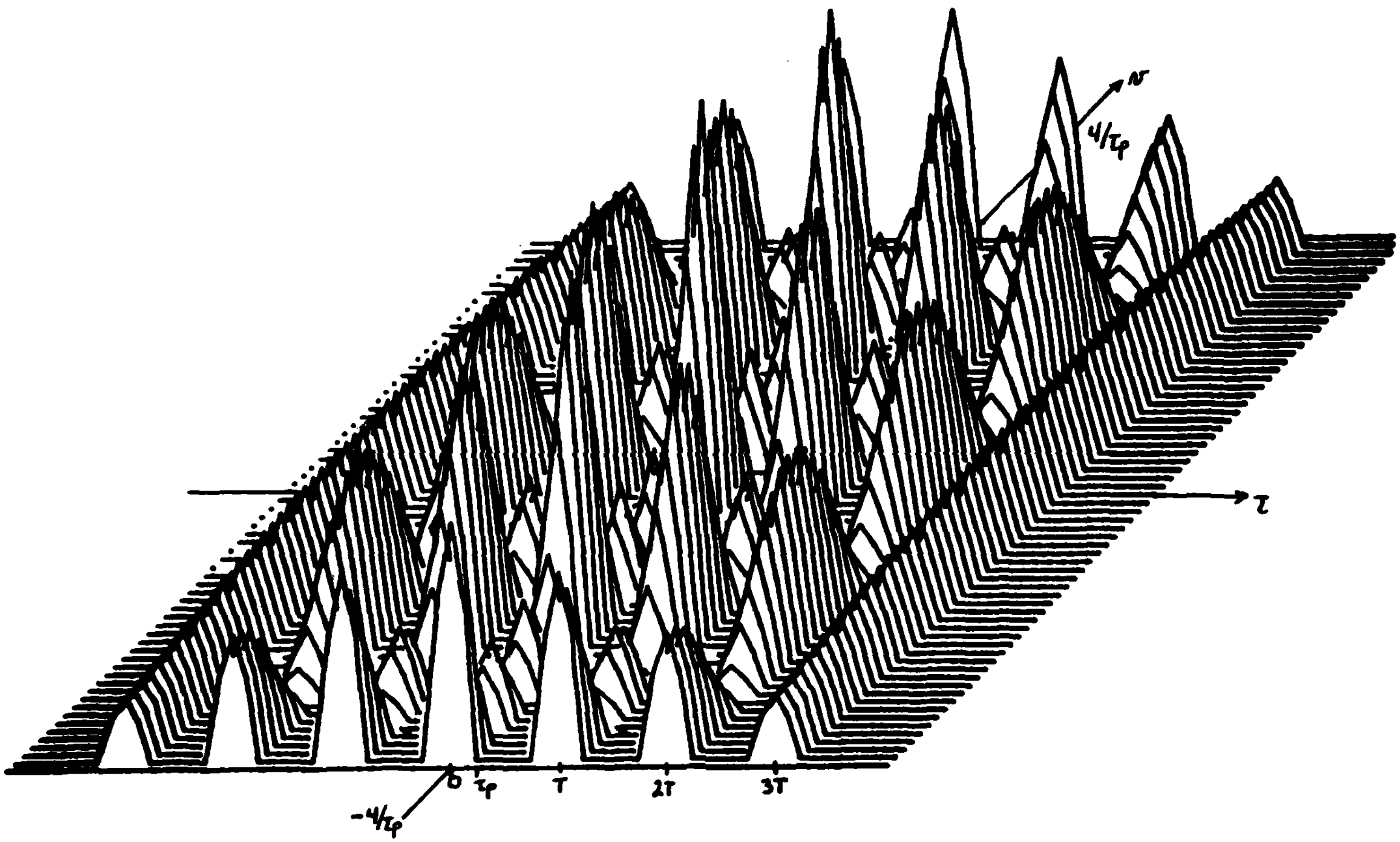
```
C*************************************************************************
C*                                                                       *
C*   SUBROUTINE TAB2                                                     *
C*                                                                       *
C*   WRITTEN BY: CAPT THOMAS L. GRIFFIN, JR                              *
C*                                                                       *
C*   26 OCTOBER 1985                                                     *
C*                                                                       *
C*   THE PURPOSE OF THIS ROUTINE IS TO CREATE A TABLE OF THE VALUES      *
C*   COMPUTED IN THE PROGRAM "CONPULS".  THIS ROUTINE WILL TABULATE      *
C*   THE VALUES STORED IN ARRAY ANSWER (ANS), WHICH IS 64X64, REAL       *
C*   VALUES.  THE ROUTINE WILL TABULATE ONLY ONE ROW OR ONE COLUMN AT    *
C*   A TIME.                                                             *
C*                                                                       *
C*   THE RDOS COMMAND TO COMPILE THE SOURCE CODE IS: FORTAN TAB2         *
C*                                                                       *
C*************************************************************************

      SUBROUTINE TAB2(ANS,TIME,WS,WIN,WF,NP,PW,PRT,RANGE,TESTP,
     *    FO,TESTF,PERCNT,PE1,PE2,FE1,FE2,NUM)

      INTEGER TAB1,ROW,DB3,MARK,MARK1,COLUMN,MAX,REPEAT,
     *        HC,NP,RANGE,TESTP,TESTF,NUM
      REAL TAU,HALF,TIME,WS,WIN,WF,PW,PRT,FO,PERCNT,PE1(16),PE2(16),
     *     FE1(16),FE2(16),ANS(64,64)



      TYPE ' '
      TYPE '***************  SUBROUTINE TAB2  ************************'
      TYPE ' '
      TYPE 'THIS ROUTINE WILL ALLOW MULTIPLE TABULATIONS OF MAGNITUDE'
      TYPE 'DATA BY ROW OR COLUMN.  AFTER EACH TABULATION, YOU WILL'
      TYPE 'BE ASKED IF YOU WANT TO TABULATE ANOTHER ROW OR COLUMN.'
      TYPE 'THE OPTION TO OBTAIN HARD COPY WILL ALSO BE PRESENTED'
      TYPE 'EACH TIME.  NOTE: THE TABLE IS ALWAYS PRESENTED TO THE'
      TYPE 'SCREEN.'
      TYPE ' '
      TYPE '***********  CAUTION  *************************************'
      TYPE ' WHEN ASKED FOR KEYBOARD INPUT, ENSURE ONLY NUMBERS'
      TYPE ' ARE ENTERED; OTHER CHARACTERS WILL CAUSE AN ERROR'
      TYPE ' RESULTING IN A RETURN TO THE SYSTEM PROMPT.  NUMBERS'
      TYPE ' ENTERED OTHER THAN THOSE REQUESTED WILL RESULT IN'
      TYPE ' UNPREDICTABLE RESULTS.  IF AN ENTRY ERROR IS MADE,'
```

```
      TYPE ' ENTER A BACKSLASH (\) FOLLOWED BY THE CORRECT ENTRY.'
      TYPE ' COMPLETE ALL ENTRIES BY PRESSING THE CARRIAGE RETURN.'
5     TYPE ' '
      TYPE 'ENTER 1 TO TABULATE A ROW, 0 FOR A COLUMN'
      ACCEPT TAB1
      TYPE ' '
      TYPE 'ENTER 1 TO OBTAIN A PRINTOUT, 0 OTHERWISE'
      ACCEPT HC

      IF(TAB1.NE.1) GO TO 10

      TYPE ' '
      TYPE 'TABULATE WHICH ROW (1-64): ENTER ROW NUMBER'
      ACCEPT ROW

      WRITE(10,300)ROW,WS+WIN*ROW
      IF(HC.EQ.1)WRITE(12,300)ROW,WS+WIN*ROW
300   FORMAT('1',9X,'ROW #= ',I2,6X,'(DOPPLER SHIFT= ',F10.3,')')
      IF(HC.EQ.1)WRITE(12,310)
310   FORMAT('  ')

      TYPE ' '
      WRITE(10,340)
      IF(HC.EQ.1)WRITE(12,340)
340   FORMAT('0',10X,'TAU',8X,'MAGNITUDE',6X,'TAU',8X,'MAGNITUDE')
      DO 45 I=1,32
         TAU=TIME*(-128+4*I)
         TAU1=TIME*(-128+4*(I+32))
         WRITE(10,350)TAU,ANS(ROW,I),TAU1,ANS(ROW,I+32)
         IF(HC.EQ.1)WRITE(12,350)TAU,ANS(ROW,I),TAU1,ANS(ROW,I+32)
350      FORMAT(9X,F12.9,4X,F6.3,7X,F12.9,4X,F6.3)
45    CONTINUE

      CALL VARS(NP,PW,PRT,NUM,TIME,WS,WF,WIN,TESTP,RANGE,TESTF,FO,
     *     PERCNT,PE1,PE2,FE1,FE2,HC)

      TYPE 'TABULATE ANOTHER ROW OR COLUMN? IF SO,'
      TYPE 'ENTER 1 NOW, 0 OTHERWISE.'
      ACCEPT REPEAT
      IF(REPEAT.EQ.1)GO TO 5
      RETURN

C***  THE NEXT SECTION OF CODE TABULATES COLUMN DATA
```

```
10      TYPE ' '
        TYPE 'TABULATE WHICH COLUMN (1-64): ENTER COLUMN NUMBER'
        ACCEPT COLUMN

        TAU=TIME*(-128+4*COLUMN)

        WRITE(10,400)COLUMN,TAU
        IF(HC.EQ.1)WRITE(12,400)COLUMN,TAU
400     FORMAT('1',9X,'COLUMN #',I2,8X,'(TAU = ',F10.6,')')

        WRITE(10,430)
        IF(HC.EQ.1)WRITE(12,430)
430     FORMAT('0',9X,'DOPPLER SHIFT',5X,'MAGNITUDE',5X,'DOPPLER SHIFT',
     *  5X,'MAGNITUDE')

        DO 80 I=1,32
           WRITE(10,440)WS+WIN*I,ANS(I,COLUMN),WS+WIN*(I+32),
     *            ANS(I+32,COLUMN)
           IF(HC.EQ.1)WRITE(12,440)WS+WIN*I,ANS(I,COLUMN),
     *     WS+WIN*(I+32),ANS(I+32,COLUMN)
440        FORMAT( 9X,F10.3,11X,F6.3,8X,F10.3,11X,F6.3)
80      CONTINUE

        CALL VARS(NP,PW,PRT,NUM,TIME,WS,WF,WIN,TESTP,RANGE,TESTF,FO,
     *      PERCNT,PE1,PE2,FE1,FE2,HC)

        TYPE 'TABULATE ANOTHER COLUMN OR ROW? IF SO,'
        TYPE 'ENTER 1 NOW, 0 OTHERWISE'
        ACCEPT REPEAT
        IF(REPEAT.EQ.1)GOTO 5

        RETURN
        END

*****************************************************************
*                                                               *
*  REFER TO PROGRAM 'GRAPH' FOR A LISTING OF SUBROUTINE 'VARS'  *
*                                                               *
*****************************************************************
```

Appendix D

Constant Carrier Data



$|\chi(\tau,\nu)|$ for Constant Carrier, N=4, PRT=4PW

COLUMN #32 (TAU = .000000)

MAX VALUE = 199 AT W=************* (SAMPLE #32)
-3dB VALUE= 63 AT W= 294.523 (SAMPLE #35)

DOPPLER SHIFT	MAGNITUDE	DOPPLER SHIFT	MAGNITUDE
-3043.418	119	98.174	181
-2945.243	88	196.348	130
-2847.069	44	294.523	63
-2748.894	0	392.698	0
-2650.719	31	490.873	42
-2552.544	40	589.047	53
-2454.370	27	687.222	35
-2356.195	0	785.397	0
-2258.020	28	883.572	34
-2159.845	44	981.743	52
-2061.671	35	1079.919	40
-1963.496	0	1178.095	0
-1865.321	54	1276.271	59
-1767.146	114	1374.442	120
-1668.972	161	1472.618	165
-1570.797	180	1570.794	180
-1472.622	165	1668.966	161
-1374.448	120	1767.142	114
-1276.273	59	1865.317	54
-1178.098	0	1963.493	0
-1079.923	40	2061.665	35
-981.749	52	2159.841	44
-883.574	34	2258.017	28
-785.399	0	2356.192	0
-687.224	35	2454.364	27
-589.050	53	2552.540	40
-490.875	42	2650.716	31
-392.700	0	2748.892	0
-294.525	63	2847.063	44
-196.351	130	2945.239	88
-98.176	181	3043.415	119
-.001	199	3141.591	128

WAVEFORM PARAMETERS: # OF PULSES= 4 PULSEWIDTH= .001000
PRT= .004000 # OF SAMPLES= 8 SPACING= .000125000

DOPPLER PARAMETERS: 1ST FREQ= -3141.6
LAST FREQ= 3141.6 FREQ INCREMENTS= 98.17474

INITIAL PHASES: OPTION DECLINED (ALL SET TO ZERO)

CARRIER DRIFT: OPTION DECLINED (ALL SET TO ZERO)

.v.0,v.l Magnitude Data for Constant Carrier, N=4, PRT=4PW

ROW #= 32 (DOPPLER SHIFT= -.001)

MAXIMUM VALUE = 199 AT TAU= .000000 (SAMPLE # 32)
-3dB VALUE = 99 AT TAU= .001000 (SAMPLE #33)

TAU	MAGNITUDE	TAU	MAGNITUDE
-.031000000	0	.001000000	99
-.030000000	0	.002000000	0
-.029000000	0	.003000000	0
-.028000000	0	.003999997	0
-.027000000	0	.004999999	0
-.026000000	0	.005999997	0
-.024999990	24	.006999999	74
-.024000000	49	.007999998	149
-.023000000	24	.008999996	74
-.022000000	0	.009999998	0
-.021000000	0	.011000000	0
-.020000000	0	.012000000	0
-.019000000	0	.013000000	0
-.018000000	0	.014000000	0
-.017000000	49	.015000000	49
-.016000000	99	.016000000	99
-.015000000	49	.017000000	49
-.014000000	0	.018000000	0
-.013000000	0	.019000000	0
-.012000000	0	.020000000	0
-.011000000	0	.021000000	0
-.009999998	0	.022000000	0
-.008999996	74	.023000000	24
-.007999998	149	.024000000	49
-.006999999	74	.024999990	24
-.005999997	0	.026000000	0
-.004999999	0	.027000000	0
-.003999997	0	.028000000	0
-.003000000	0	.029000000	0
-.002000000	0	.030000000	0
-.001000000	99	.031000000	0
.000000000	199	.032000000	0

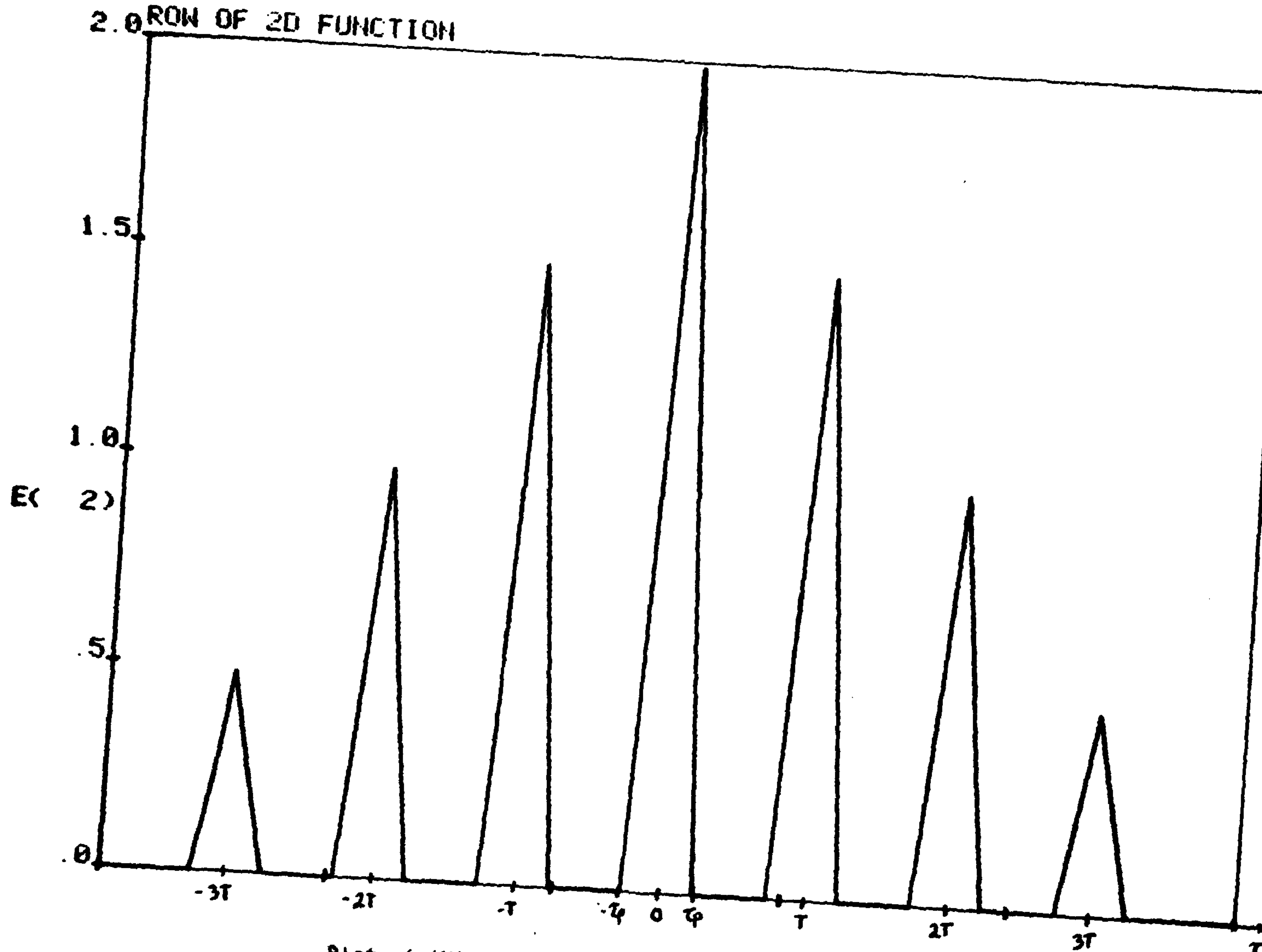
WAVEFORM PARAMETERS: # OF PULSES= 4 PULSEWIDTH= .001000
PRT= .004000 # OF SAMPLES= 8 SPACING= .000125000

DOPPLER PARAMETERS: 1ST FREQ= -3141.6
LAST FREQ= 3141.6 FREQ INCREMENTS= 98.17474

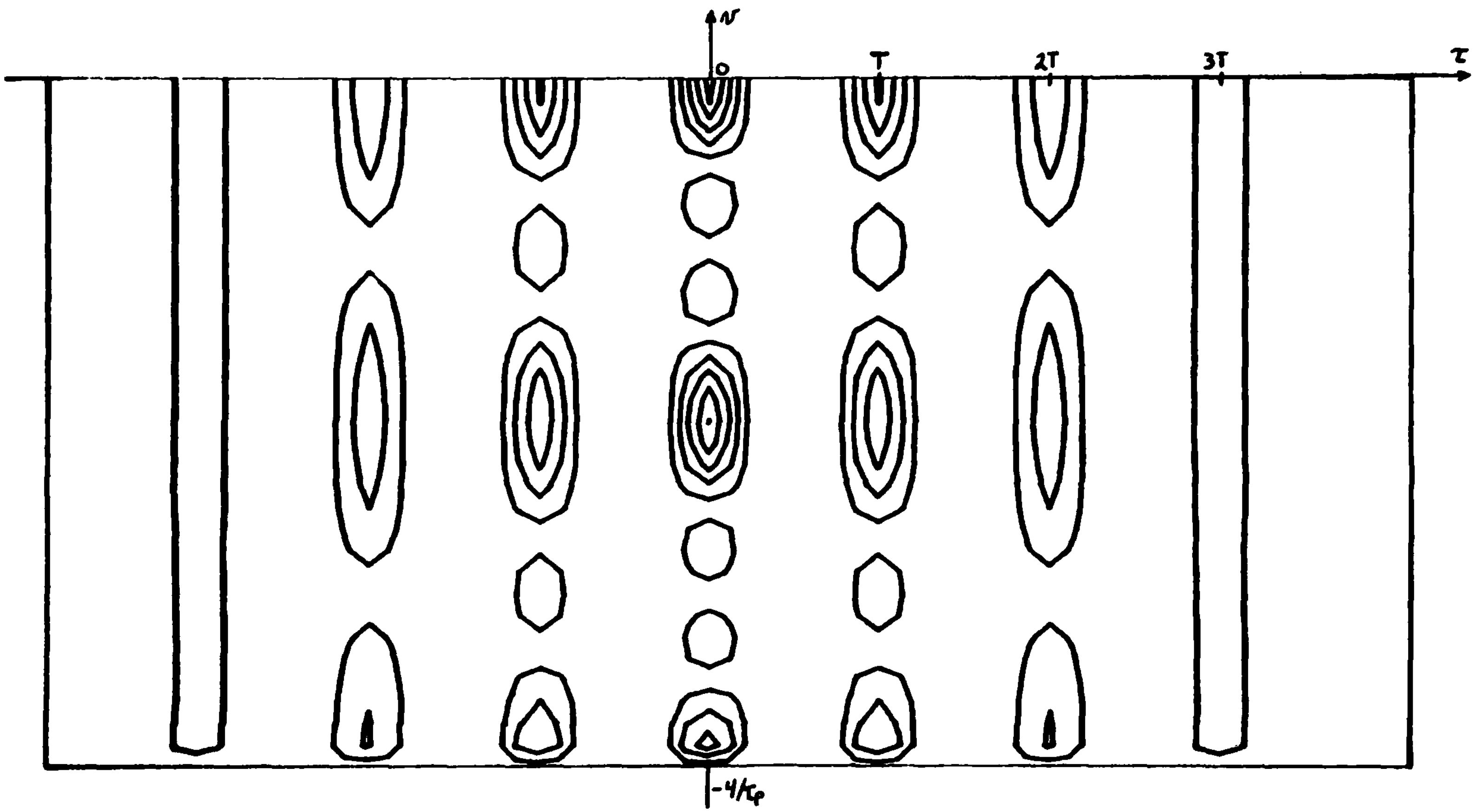
INITIAL PHASES: OPTION DECLINED (ALL SET TO ZERO)

CARRIER DRIFT: OPTION DECLINED (ALL SET TO ZERO)

[illegible] Magnitude Data for Constant Carrier, N=4, PRT=4PW



Plot of $|X(\tau,0)|$ for Constant Carrier, N=4, PRT=4PW



5 Level Contour of $|\chi(\tau,\nu)|$ for Constant Carrier,
N=4, PRT=4PW

ROW #= 32 (DOPPLER SHIFT= -.001)

MAXIMUM VALUE = 199 AT TAU= .000000 (SAMPLE # 32)
-3dB VALUE = 99 AT TAU= .001000 (SAMPLE #34)

TAU	MAGNITUDE	TAU	MAGNITUDE
-.015500000	0	.000500000	149
-.015000000	0	.001000000	99
-.014500000	0	.001500000	49
-.014000000	0	.002000000	0
-.013500000	0	.002500000	0
-.013000000	0	.003000000	0
-.012500000	0	.003500000	0
-.012000000	0	.003999997	0
-.011500000	0	.004499998	0
-.011000000	0	.004999999	0
-.010500000	0	.005499996	0
-.009999998	0	.005999997	0
-.009499997	24	.006499998	24
-.008999996	49	.006999999	49
-.008499999	74	.007499997	74
-.007999998	99	.007999998	99
-.007499997	74	.008499999	74
-.006999999	49	.008999996	50
-.006499998	24	.009499997	25
-.005999997	0	.009999998	0
-.005499996	0	.010500000	0
-.004999999	0	.011000000	0
-.004499998	0	.011500000	0
-.003999997	0	.012000000	0
-.003500000	0	.012500000	0
-.003000000	0	.013000000	0
-.002500000	0	.013500000	0
-.002000000	0	.014000000	0
-.001500000	49	.014500000	0
-.001000000	99	.015000000	0
-.000500000	149	.015500000	0
.000000000	199	.016000000	0

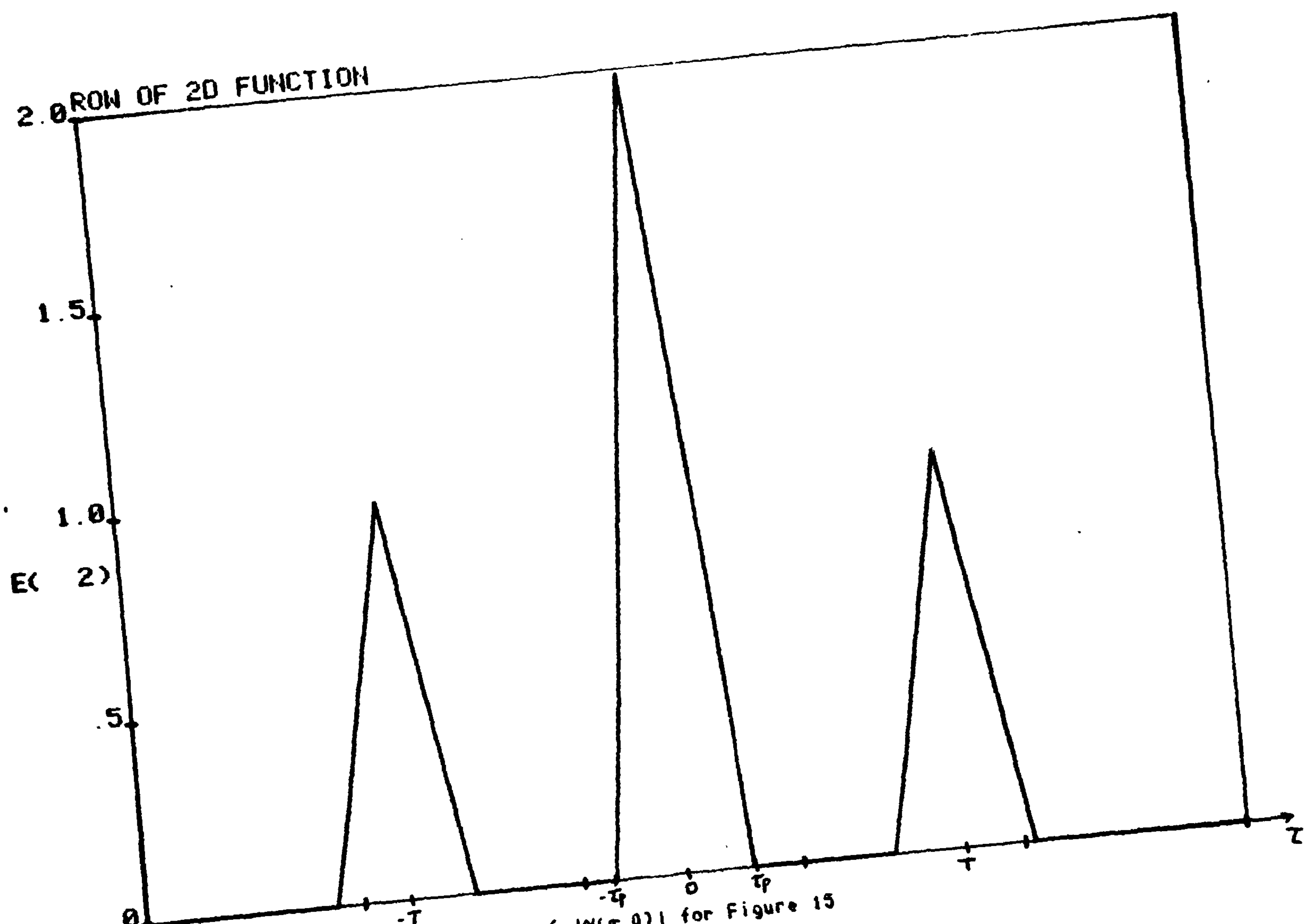
WAVEFORM PARAMETERS: # OF PULSES= 2 PULSEWIDTH= .001000
PRT= .004000 # OF SAMPLES= 16 SPACING= .000062500

DOPPLER PARAMETERS: 1ST FREQ= -3141.6
LAST FREQ= 3141.6 FREQ INCREMENTS= 98.17474

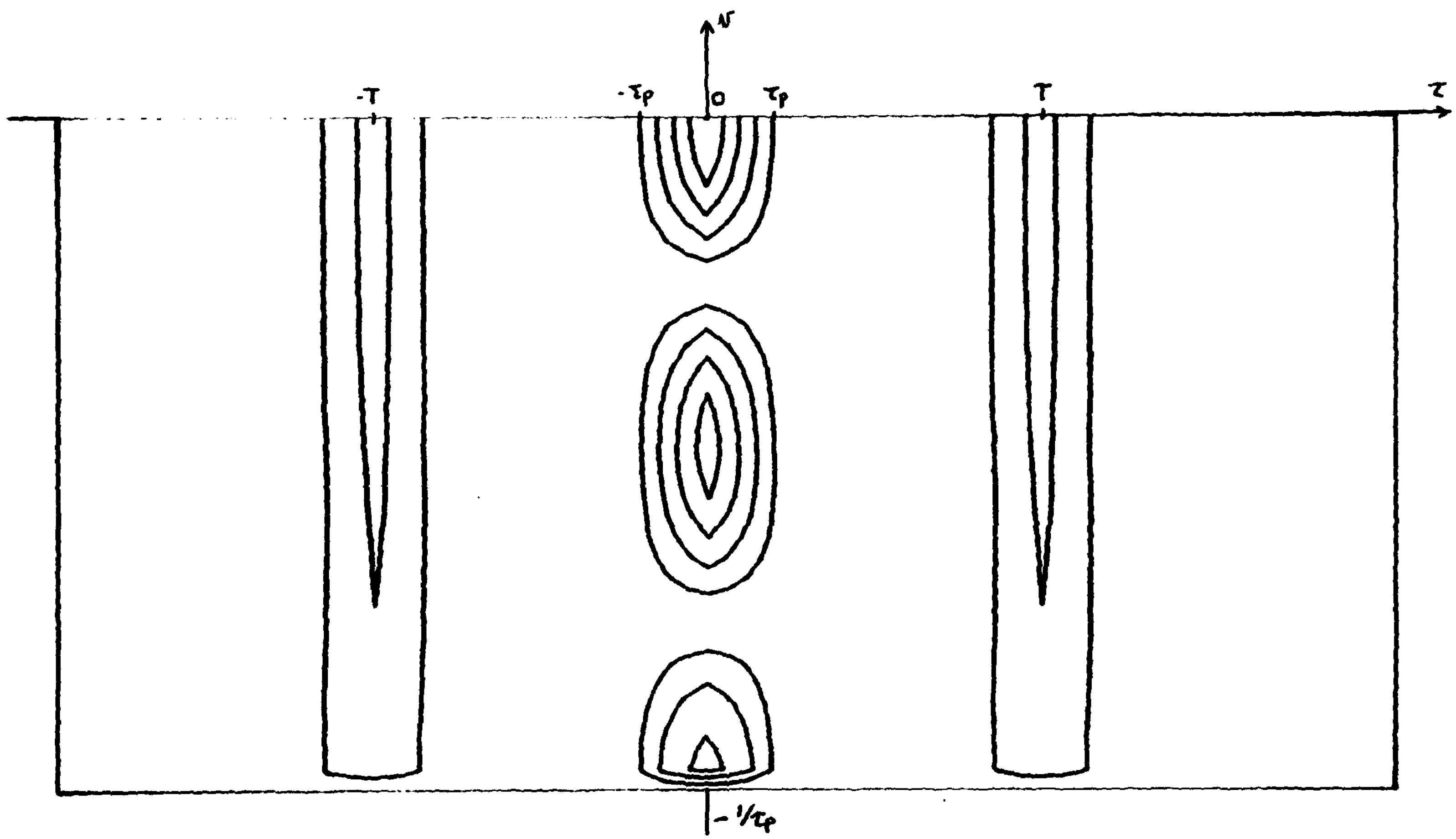
INITIAL PHASES: OPTION DECLINED (ALL SET TO ZERO)

CARRIER DRIFT: OPTION DECLINED (ALL SET TO ZERO)

$|X(\tau,0)|$ Magnitude Data of Figure 15



Plot of $|X(\tau,0)|$ for Figure 15



5 Level Contour for Figure 15

Appendix E

Source Data for Figure 16

COLUMN #32 (TAU = .000000)

MAX VALUE = 199 AT W= -.000 (SAMPLE #32)
-3dB VALUE= 27 AT W= 859.027 (SAMPLE #33)

DOPPLER SHIFT	MAGNITUDE	DOPPLER SHIFT	MAGNITUDE
-26629.910	16	859.027	27
-25770.880	1	1718.059	140
-24911.850	1	2577.086	40
-24052.820	2	3436.117	36
-23193.790	1	4295.145	5
-22334.760	3	5154.176	5
-21475.730	6	6013.203	3
-20616.700	13	6872.234	4
-19757.680	2	7731.262	30
-18898.640	0	8590.293	4
-18039.620	0	9449.320	44
-17180.590	19	10308.350	6
-16321.560	7	11167.380	18
-15462.530	14	12026.410	2
-14603.500	4	12885.440	1
-13744.470	0	13744.470	0
-12885.440	1	14603.500	4
-12026.410	2	15462.530	14
-11167.380	18	16321.550	7
-10308.350	6	17180.590	19
-9449.324	44	18039.610	0
-8590.293	4	18898.640	0
-7731.266	30	19757.670	2
-6872.234	4	20616.700	13
-6013.207	3	21475.730	6
-5154.176	5	22334.760	3
-4295.148	5	23193.790	1
-3436.117	36	24052.820	2
-2577.090	40	24911.850	1
-1718.059	140	25770.880	1
-859.031	27	26629.910	16
.000	199	27488.940	0

WAVEFORM PARAMETERS: # OF PULSES= 4 PULSEWIDTH= .001000
PRT= .004000 # OF SAMPLES= 8 SPACING= .000125000

DOPPLER PARAMETERS: 1ST FREQ= -27488.9
LAST FREQ= 27488.9 FREQ INCREMENTS= 859.02930

INITIAL PHASES: OPTION DECLINED (ALL SET TO ZERO)

CARRIER DRIFT: OPTION DECLINED (ALL SET TO ZERO)

[illegible] Magnitude Data for Figure 16

****** DATA FOR ROW 32 *****

MAXIMUM VALUE= 32.000 AT TAU= .00000 (SAMPLE 128)
-3DB VALUE= .000 AT TAU= .00025 (SAMPLE 130)

1-32	33-64	65-96	97-128	129-160	161-192	193-224	225-256
.000	1.000	1.848	2.414	18.292	2.414	1.848	1.000
.000	.765	2.000	1.848	.000	1.848	2.000	.765
.000	.414	.765	1.000	5.412	1.000	.766	.414
.000	1.082	.000	2.613	.000	2.613	.000	1.082
.000	.414	.765	1.000	3.247	1.000	.765	.414
.000	.765	2.000	1.848	.000	1.847	2.000	.765
.000	1.000	1.848	2.414	2.613	2.414	1.848	1.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000	.000
1.000	1.848	2.414	2.613	2.414	1.848	1.000	.000
.765	2.000	1.848	.000	1.848	2.000	.765	.000
.414	.765	1.000	3.247	1.000	.766	.414	.000
1.082	.000	2.613	.000	2.613	.000	1.082	.000
.414	.765	1.000	5.412	1.000	.765	.414	.000
.765	2.000	1.848	.000	1.848	2.000	.765	.000
1.000	1.848	2.414	18.292	2.414	1.848	1.000	.000
.000	.000	.000	32.000	.000	.000	.000	.000

$|X(\tau,0)|$ Magnitude Data for Figure 16

Vita

Captain Thomas L. Griffin Jr. was born in 1947 in Oakland, CA. After enlisting in the US Air Force in 1966, he served as an aircraft avionics technician from 1967 to 1974. From 1975 to 1978 he served as an avionics instructor during which time he earned an Associate of Arts from the Community College of the Air Force in avionics radar technology. He entered an education and commissioning program in 1979, graduated from San Jose State University, CA with a BSEE in May 1981, and received a commission as a 2Lt in Oct 1981. After serving as a project engineer for weather equipment acquisition with Headquarters Air Weather Service, he entered the MSEE program at the Air Force Institute of Technology, WPAFB OH in 1984. Specializing in communications and radar, he will complete the MSEE requirements in Dec 1985.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

AD-A163 953

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION: UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT: Approved for public release; distribution unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S): AFIT/GE/ENG/85D-19	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION: School of Engineering; 6b. OFFICE SYMBOL (If applicable): AFIT/ENG	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS (City, State and ZIP Code): Air Force Institute of Technology, Wright-Patterson AFB, Ohio 45433	7b. ADDRESS (City, State and ZIP Code)
8a. NAME OF FUNDING/SPONSORING ORGANIZATION: AF Wright Aeronautical Lab; 8b. OFFICE SYMBOL (If applicable): AFWAL/AARM-2	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State and ZIP Code): Wright-Patterson AFB, Ohio 45433	10. SOURCE OF FUNDING NOS.: PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT NO.
11. TITLE (Include Security Classification): See Box 19	

12. PERSONAL AUTHOR(S): Thomas L Griffin, Jr., B.S.E.E., Capt, UASF

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Yr., Mo., Day)	15. PAGE COUNT
MS Thesis	FROM ____ TO ____	1985 December	168

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB. GR.	
17	09		Radar Ambiguity Function, Step Frequency Pulsetrains, Radar Resolution

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Title: EFFECTS OF RANDOM PHASE SHIFTS AND CARRIER DRIFT ON THE RESOLUTION PROPERTIES OF LINEAR STEP FREQUENCY PULSE TRAINS

Thesis Chairman: Dr. Vittal Pyati
Associate Professor of Electrical Engineering

Approved for public release: IAW AFR 190-1.
LYNN E. WOLAVER 16 JAN 86
Dean for Research and Professional Development
Air Force Institute of Technology (AFC)
Wright-Patterson AFB OH 45433

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT: UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS ☐	21. ABSTRACT SECURITY CLASSIFICATION: UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL: Dr. Vittal Pyati, Ph.D	22b. TELEPHONE NUMBER (Include Area Code): 513-255-3576; 22c. OFFICE SYMBOL: AFIT/ENG

DD FORM 1473, 83 APR — EDITION OF 1 JAN 73 IS OBSOLETE.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

A FORTRAN program using fast Fourier transforms has been written to calculate and plot the ambiguity function of a uniform linear step frequency pulse train with random initial phases and carrier frequency drifts. The effects of random phases and frequency drifts are studied in two stages. First, the initial phases are allowed to assume any value with equal probability in ranges of 0 to 20 degrees and 0 to 180 degrees while the frequency drifts are zero. Secondly, the phases are fixed at zero and carrier frequency drifts up to 0.1% and 1.0% are set. In each case, the ambiguity surface is shifted along the doppler axis with the effect being more pronounced for carrier frequency drifts. The effects on the half power width of the central lobe for 10 test cases for each phase and frequency range appear negligible for a two pulse train.

END

FILMED

3-86

DTIC